speech understanding systems," Rep. R-1434-ARPA, June 1974 (available from ARPA under Order 189-1).
[26] M. W. Grady and M. B. Herscher, "Advanced speech technology applied to problems of air traffic control," in *NAECON 1975 Conf. Rec.*, Dayton, OH, pp. 541-546, June 1975.
[27] D. Hill and E. Wacker, "ESOTERIC-II—An approach to practical voice control: progress report," *Machine Intelligence*, Edinburgh, Scotland: Edinburgh Univ. Press, 1969, vol. 5, pp. 463-493.
[28] M. Medress, "A procedure for machine recognition of speech," in *Conf. Rec., 1972 Conf. Speech Communication and Processing*, pp. 113-116 (Newton, MA, Apr. 1972, (AD-742236)).
[29] P. B. Scott, "Voice input code identifier," Final Tech. Rep.—1975, Rome Air Development Center, Air Force Systems Command, Griffiss AFB, NY.
[30] F. Itakura, "Minimum prediction residual applied to speech recognition," *IEEE Trans. Acoust. Speech Signal Processing*, vol. ASSP-23, pp. 67-72, Feb. 1975.
[31] T. B. Martin, H. J. Zadell, E. F. Grunza, and M. B. Herscher, "Numeric speech translating system," in *Automatic Pattern Recognition*, pp. 113-141, May 1969, Washington, DC: Nat. Security Industiral Ass.
[32] M. R. Sambur and L. R. Rabiner, "A speaker independent digit recognition system," *Bell Syst. Tech. J.*, vol. 54, Jan. 1975.
[33] D. G. Bobrow and D. H. Klatt, "A limited speech recognition system," BBN Rep. 1667, Final Rep., Contract NAS 12-138, Bolt, Beranek and Newman, Inc., May 15, 1975.
[34] J. N. Shearme and P. F. Leach, "Some experiments with a simple word recognition system," *IEEE Trans. Audio Electroacoust.*, vol. AU-16, pp. 256-261, 1967.
[35] G. L. Clapper, "Automatic word recognition," *IEEE Spectrum*, vol. 16, pp. 57-69, 1971.
[36] V. M. Velichiko and N. G. Zagoruiko, "Automatic recognition of 200 words," *Int. J. Man-Machine Studies*, vol. 2, p. 223, 1970.
[37] K. Kido, H. Suzuki, S. Makino, and T. Matsouka, "Recognition of spoken words by use of spectral peaks and lexicon," paper presented at IEEE Symp. Speech Recognition, Carnegie-Mellon Univ., Pittsburgh, PA, Apr. 15-19, 1974.
[38] A. Ichikawa, Y. Nakano, and K. Nakata, "Evaluation of various parameter sets in spoken digits recognition," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 202-209, June 1973.
[39] T. G. Von Keller, "An on-line recognition system for spoken digits," *J. Acoust. Soc. Amer.*, pp. 1288-1296, 1971.
[40] C. F. Teacher, H. G. Kellet, and L. R. Focht, "Experimental, limited-vocabulary speech recognizer," *IEEE Trans. Audio Electroacoust.*, vol. AU-15, pp. 127-130, 1967.
[41] L. C. W. Pols, "Real-time recognition of spoken words," *IEEE Trans. Computers*, vol. C-20, pp. 972-978, 1971.
[42] B. Gold, "Word-recognition computer program," MIT Res. Lab. *Electronics*, Cambridge, MA, Tech. Rep. 452, June 15, 1966.
[43] R. DeMori, L. Gilli, and A. R. Meo, "A flexible real-time recognizer of spoken words for man-machine communication," *Int. J. Man-Machine Studies*, vol. 2, pp. 317-326, 1970.
[44] J. W. Glenn, "Machines you can talk to," *Machine Design*, pp. 72-75, May 1, 1975.

Speech Recognition by Machine: A Review

D. RAJ REDDY

Abstract—**This paper provides a review of recent developments in speech recognition research. The concept of sources of knowledge is introduced and the use of knowledge to generate and verify hypotheses is discussed. The difficulties that arise in the construction of different types of speech recognition systems are discussed and the structure and performance of several such systems is presented. Aspects of component subsystems at the acoustic, phonetic, syntactic, and semantic levels are presented. System organizations that are required for effective interaction and use of various component subsystems in the presence of error and ambiguity are discussed.**

I. Introduction

THE OBJECT of this paper is to review recent developments in speech recognition. The Advanced Research Projects Agency's support of speech understanding research has led to a significantly increased level of activity in this area since 1971. Several connected speech recognition systems have been developed and demonstrated. The role and use of knowledge such as acoustic-phonetics, syntax, semantics, and context are more clearly understood. Computer programs for speech recognition seem to deal with ambiguity, error, and nongrammaticality of input in a graceful and effective manner that is uncommon to most other computer programs. Yet there is still a long way to go. We can handle relatively restricted task domains requiring simple grammatical structure and a few hundred words of vocabulary for single trained speakers in controlled environments, but we are very far from being able to handle relatively unrestricted dialogs from a large population of speakers in uncontrolled environments. Many more years of intensive research seem necessary to achieve such a goal.

Sources of Information: The primary sources of information in this area are the *IEEE Transactions on Acoustics, Speech, and Signal Processing* (pertinent special issues: vol. 21, June 1973; vol. 23, Feb. 1975) and the *Journal of the Acoustical Society of America* (in particular, Semiannual Conference Abstracts which appear with January and July issues each year; recently they have been appearing as spring and fall supplements). Other relevant journals are *IEEE Transactions* (Computer; Information Theory; and Systems, Man, and

Manuscript received September 1, 1975; revised November 19, 1975. This work was supported in part by the Advanced Research Projects Agency and in part by the John Simon Guggenheim Memorial Foundation.

The author is with the Computer Science Department, Carnegie-Mellon University, Pittsburgh, PA 15213.

Cybernetics), *Communications of ACM, International Journal of Man-Machine Studies, Artificial Intelligence,* and *Pattern Recognition.*

The books by Flanagan [44], Fant [40], and Lehiste [84] provide extensive coverage of speech, acoustics, and phonetics, and form the necessary background for speech recognition research. Collections of papers, in the books edited by David and Denes [25], Lehiste [83], Reddy [121], and Wathen-Dunn [158], and in conference proceedings edited by Erman [34] and Fant [41], provide a rich source of relevant material. The articles by Lindgren [88], Hyde [66], Fant [39], Zagoruiko [171], Derkach [27], Hill [63], and Otten [113] cover the research progress in speech recognition prior to 1970 and proposals for the future. The papers by Klatt [74] and Wolf [163] provide other points of view of recent advances.

Other useful sources of information are research reports published by various research groups active in this area (and can be obtained by writing to one of the principal researchers given in parentheses): Bell Telephone Laboratories (Denes, Flanagan, Fujimura, Rabiner); Bolt Beranek and Newman, Inc. (Makhoul, Wolf, Woods); Carnegie-Mellon University (Erman, Newell, Reddy); Department of Speech Communication, KTH, Stockholm (Fant); Haskins Laboratories (Cooper, Mermelstein); IBM Research Laboratories (Bahl, Dixon, Jelinek); M.I.T. Lincoln Laboratories (Forgie, Weinstein); Research Laboratory of Electronics, M.I.T. (Klatt); Stanford Research Institute (Walker); Speech Communication Research Laboratory (Broad, Markel, Shoup); System Development Corporation (Barnett, Ritea); Sperry Univac (Lea, Medress); University of California, Berkeley (O'Malley); Xerox Palo Alto Research Center (White); and Threshold Technology (Martin). In addition there are several groups in Japan and Europe who publish reports in national languages and English. Complete addresses for most of these groups can be obtained by referring to author addresses in the *IEEE Trans. Acoust., Speech, Signal Processing*, June 1973 and Feb. 1975. For background and introductory information on various aspects of speech recognition we recommend the tutorial-review papers on "Speech understanding systems" by Newell, "Parametric representations of Speech" by Schafer and Rabiner, "Linear prediction in automatic speech recognition" by Makhoul, "Concepts for Acoustic-Phonetic recognition" by Broad and Shoup, "Syntax, Semantics and Speech" by Woods, and "System organization for speech understanding" by Reddy and Erman, all appearing in *Speech Recognition: Invited Papers of the IEEE Symposium* [121].

Scope of the Paper: This paper is intended as a review and not as an exhaustive survey of all research in speech recognition. It is hoped that, upon reading this paper, the reader will know what a speech recognition system consists of, what makes speech recognition a difficult problem, and what aspects of the problem remain unsolved. To this end we will study the structure and performance of some typical systems, component subsystems that are needed, and system organization that permits effective interaction and use of the components. We do not attempt to give detailed descriptions of systems or mathematical formulations, as these are available in published literature. Rather, we will mainly present distinctive and novel features of selected systems and their relative advantages.

Many of the comments of an editorial nature that appear in this paper represent one point of view and are not necessarily shared by all the researchers in the field. Two other papers appearing in this issue, Jelinek's on statistical approaches and Martin's on applications, augment and complement this paper. Papers by Flanagan and others, also appearing in this issue, look at the total problem of man-machine communication by voice.

A. The Nature of the Speech Recognition Problem

The main goal of this area of research is to develop techniques and systems for speech input to machines. In earlier attempts, it was hoped that learning how to build simple recognition systems would lead in a natural way to more sophisticated systems. Systems were built in the 1950's for vowel recognition and digit recognition, producing creditable performance. But these techniques and results could not be extended and extrapolated toward larger and more sophisticated systems. This had led to the appreciation that linguistic and contextual cues must be brought to bear on the recognition strategy if we are to achieve significant progress. The many dimensions that affect the feasibility and performance of a speech recognition system are clearly stated in Newell [108].

Fig. 1 characterizes several different types of speech recognition systems ordered according to their intrinsic difficulty. There are already several commercially available isolated word recognition systems today. A few research systems have been developed for restricted connected speech recognition and speech understanding. There is hope among some researchers that, in the not too distant future, we may be able to develop interactive systems for taking dictation using a restricted vocabulary. Unlimited vocabulary speech understanding and connected speech recognition systems seem feasible to some, but are likely to require many years of directed research.

The main feature that is used to characterize the complexity of a speech recognition task is whether the speech is connected or is spoken one word at a time. In connected speech, it is difficult to determine where one word ends and another begins, and the characteristic acoustic patterns of words exhibit much greater variability depending on the context. *Isolated word recognition systems* do not have these problems since words are separated by pauses.

The second feature that affects the complexity of system is the vocabulary size. As the size or the confusability of a vocabulary increases, simple brute-force methods of representation and matching become too expensive and unacceptable. Techniques for compact representation of acoustic patterns of words, and techniques for reducing search by constraining the number of possible words that can occur at a given point, assume added importance.

Just as vocabulary is restricted to make a speech recognition problem more tractable, there are several other aspects of the problem which can be used to constrain the speech recognition task so that what might otherwise be an unsolvable problem becomes solvable. The rest of the features in Fig. 1, i.e., task-specific knowledge, language of communication, number and cooperativeness of speakers, and quietness of environment, represent some of the commonly used constraints in speech recognition systems.

One way to reduce the problems of error and ambiguity resulting from the use of connected speech and large vocabularies is to use all the available task-specific information to reduce search. The *restricted speech understanding systems* (Fig. 1, line 3) assume that the speech signal does not have all the necessary information to uniquely decode the message and

	Mode of Speech	Vocabulary Size	Task Specific Information	Language	Speaker	Environment
Word recognition-isolated (WR)	isolated words	10-300	limited use	—	cooperative	—
Connected speech recognition-restricted (CSR)	connected speech	30-500	limited use	restricted command language	cooperative	quiet room
Speech understanding-restricted (SU)	connected speech	100-2000	full use	English-like	not uncooperative	—
Dictation machine-restricted (DM)	connected speech	1000-10000	limited use	English-like	cooperative	quiet room
Unrestricted speech understanding (USU)	connected speech	unlimited	full use	English	not uncooperative	—
Unrestricted connected speech recognition (UCSR)	connected speech	unlimited	none	English	not uncooperative	quiet room

Fig. 1. Different types of speech recognition systems ordered according to their intrinsic difficulty, and the dimensions along which they are usually constrained. Vocabulary sizes given are for some typical systems and can vary from system to system. It is assumed that a cooperative speaker would speak clearly and would be willing to repeat or spell a word. A not uncooperative speaker does not try to confuse the system but does not want to go out of his way to help it either. In particular, the system would have to handle "uhms" and "ahs" and other speech-like noise. The "—" indicates an "unspecified" entry variable from system to system.

that, to be successful, one must use all the available sources of knowledge to infer (or deduce) the intent of the message [107]. The performance criterion is somewhat relaxed in that, as long as the message is understood, it is not important to recognize each and every phoneme and/or word correctly. The requirement of using all the sources of knowledge, and the representation of the *task*, *conversational context*, *understanding*, and *response generation*, all add to the difficulty and overall complexity of speech understanding systems.

The *restricted connected speech recognition systems* (Fig. 1, line 2) keep their program structure simple by using only some task-specific knowledge, such as restricted vocabulary and syntax, and by requiring that the speaker speak clearly and use a quiet room. The simpler program structure of these systems provides an economical solution in a restricted class of connected speech recognition tasks. Further, by not being task-specific, they can be used in a wider variety of applications without modification.

The restricted speech understanding systems have the advantage that by making effective use of all the available knowledge, including semantics, conversational context, and speaker preferences, they can provide a more flexible and hopefully higher performance system. For example, they usually permit an English-like grammatical structure, do not require the speaker to speak clearly, and permit some nongrammaticality (including babble, mumble, and cough). Further, by paying careful attention to the task, many aspects of error detection and correction can be handled naturally, thus providing a graceful interaction with the user.

The (*restricted*) *dictation machine* problem (Fig. 1, line 4) requires larger vocabularies (1000 to 10 000 words). It is assumed that the user would be willing to spell any word that is unknown to the system. The task requires an English-like syntax, but can assume a cooperative speaker speaking clearly in a quiet room.

The *unrestricted speech understanding* problem requires unlimited vocabulary connected speech recognition, but permits the use of all the available task-specific information. The most difficult of all recognition tasks is the *unrestricted connected speech recognition* problem which requires unlimited vocabulary, but does not assume the availability of any task-specific information.

We do not have anything interesting to say about the last three tasks, except perhaps speculatively. In Section II, we will study the structure and performance of several systems of the first three types (Fig. 1), i.e., isolated word recognition systems, restricted connected speech recognition systems, and restricted speech understanding systems.

In general, for a given system and task, performance depends on the size and speed of the computer and on the accuracy of the algorithm used. Accuracy is often task dependent. (We shall see in Section II that a system which gives 99-percent accuracy on a 200-word vocabulary might give only 89-percent accuracy on a 36-word vocabulary.) Accuracy versus response time tradeoff is also possible, i.e., it is often possible to tune a system and adjust thresholds so as to improve the response time while reducing accuracy and vice versa.

Sources of Knowledge: Many of us are aware that a native speaker uses, subconsciously, his knowledge of the language, the environment, and the context in understanding a sentence. These sources of knowledge (KS's) include the characteristics of speech sounds (*phonetics*), variability in pronunciations (*phonology*), the stress and intonation patterns of speech (*prosodics*), the sound patterns of words (*lexicon*), the grammatical structure of language (*syntax*), the meaning of words and sentences (*semantics*), and the context of conversation (*pragmatics*). Fig. 2 shows the many dimensions of variability of these KS's; it is but a slight reorganization (to correspond to the sections of this paper) of a similar figure appearing in [108].

1.	Performance	Nature of input	Isolated words? connected speech?
		Response time	Real time? close to real-time? no hurry?
		Accuracy	Error-free (>99.9%)? almost error-free (>99%)? occasional error (>90%)?
2.	Source characteristics (acoustic knowledge)	Acoustic analysis	Airconditioning noise? computer room? reverberation?
		Noise sources	Dialect? sex? age? cooperative?
		Speaker characteristics	High quality microphone? telephone? Spectrum? formants? zerocrossings? LPC?
3.	Language characteristics (phonetic knowledge)	Features	Voiced? energy? stress? intonation?
		Phones	Number? distinguishability?
		Phonology	Phone realization rules? junction rules?
		Word realization	Insertion, deletion and change rules? Word hypothesis? word verification?
4.	Problem characteristics (task specific knowledge)	Size of vocabulary	10? 100? 1,000? 10,000?
		Confusability of vocabulary	High? what equivalent vocabulary?
		Syntactic support	Artificial language? free English?
		Semantic and contextual support	Constrained task? open semantics?
5.	System characteristics	Organization	Strategy? representation?
		Interaction	Graceful interaction with user? graceful error recovery?

Fig. 2. Factors affecting feasibility and performance of speech recognition systems. (Adapted from Newell *et al.* [108].)

To illustrate the effect of some of these KS's, consider the following sentences.

1) Colorless paper packages crackle loudly.
2) Colorless yellow ideas sleep furiously.
3) Sleep roses dangerously young colorless.
4) Ben burada ne yaptigimi bilmiyorum.

The first sentence, though grammatical and meaningful, is pragmatically implausible. The second is syntactically correct but meaningless. The third is both syntactically and semantically unacceptable. The fourth (a sentence in Turkish) is completely unintelligible to most of us. One would expect a listener to have more difficulty in recognizing a sentence if it is inconsistent with one or more KS's. Miller and Isard [101] show that this is indeed the case.

If the knowledge is incomplete or inaccurate, people will tend to make erroneous hypotheses. This can be illustrated by a simple experiment. Subjects were asked to listen to two sentences and write down what they heard. The sentences were "In mud eels are, in clay none are" and "In pine tar is, in oak none is." The responses of four subjects are given below.

In mud eels are,	*In clay none are*
in muddies sar	in clay nanar
in my deals are	en clainanar
in my ders	en clain
in model sar	in claynanar
In pine tar is,	*In oak none is*
in pine tarrar	in oak ? es
in pyntar es	in oak nonnus
in pine tar is	in ocnonin
en pine tar is	in oak is

The responses show that the listener forces his own interpretation of what he hears, and not necessarily what may have been intended by the speaker. Because the subjects do not have the contextual framework to expect the words "mud eels" together, they write more likely sounding combinations such as "my deals" or "models." We find the same problem with words such as "oak none is." Notice that they failed to detect where one word ends and another begins. It is not uncommon for machine recognition systems to have similar problems with word segmentation. To approach human performance, a machine must also use all the available KS's effectively.

Reddy and Newell [124] show that knowledge at various levels can be further decomposed into sublevels (Fig. 3) based on whether it is task-dependent *a priori* knowledge, conversation-dependent knowledge, speaker-dependent knowledge, or analysis-dependent knowledge. One can further decompose each of these sublevels into sets of rules relating to specific topics. Many of the present systems have only a small subset of all the KS's shown in Fig. 3. This is because much of this knowledge is yet to be identified and codified in ways that can be conveniently used in a speech understanding system. Sections III through V review the recent progress in representation and use of various sources of knowledge.

In Section III, we consider aspects of signal processing for speech recognition. There is a great deal of research and many publications in this area, but very few of them are addressed to questions that arise in building speech recognition systems. It is not uncommon for a speech recognition system to show a catastrophic drop in performance when the microphone is changed or moved to a slightly noisy room. Many parametric representations of speech have been proposed but there are few comparative studies. In Section III, we shall review the techniques that are presently used in speech signal and analysis and noise normalization, and examine their limitations.

There are several KS's which are common to most connected speech recognition systems and independent of the task. These can be broadly grouped together as task-independent aspects of a speech recognition system. Topics such as feature extraction, phonetic labeling, phonological rules, (bottom-up) word hypothesis, and word verification fall into this category. In Section IV, we will review the techniques used and the present state of accomplishment in these areas.

Given a task that is to be performed using a speech recognition system, one is usually able to specify the vocabulary, the grammatical structure of sentences, and the semantic and contextual constraints provided by the task. In Section V, we will discuss the nature, representation, and use of these KS's in a recognition (or understanding) system.

Control Structure and System Organization: How is a given source of knowledge used in recognition? The Shannon [140] experiment gives a clue. In this experiment, human subjects demonstrate their ability to predict (and correct) what will appear next, given a portion of a sentence.

Just as in the above experiment, many recognition systems use the KS's to generate hypotheses about what word might

Type of knowledge	Task-dependent knowledge	Conversation-dependent knowledge	Speaker-dependent knowledge	Analysis-dependent knowledge
Pragmatic and Semantic	A priori semantic knowledge about the task domain	Concept subselection based on conversation	Psychological model of the user	Concept subselection based on partial sentence recognition
Syntactic	Grammar for the language	Grammar subselection based on topic	Grammar subselection based on speaker	Grammar subselection based on partial phrase recognition
Lexical	Size and confusability of the vocabulary	Vocabulary subselection based on topic	Vocabulary subselection and ordering based on speaker preference	Vocabulary subselection based on segmental features
Phonemic and phonetic	Characteristics of phones and phonemes of the language	Contextual variability in phonemic characteristics	Dialectal variations of the speaker	Phonemic subselection based on segmental features
Parametric and acoustic	A priori knowledge about the transducer characteristics	Adaptive noise normalization	Variations resulting from the size and shape of vocal tract	Parameter tracking based on previous parameters

Fig. 3. Sources of knowledge (KS). (From Reddy and Newell [124].)

(1) Speed of Communication	Speech is about 4 times faster than standard manual input for continuous text.
(2) Total System Response Time	Direct data entry from remote source, which avoids relayed entry via intermediate human transducers, speeds up communication substantially.
(3) Total System Reliability	Direct data entry from remote source with immediate feedback, avoiding relayed entry via intermediate human transducers, increases reliability substantially.
(4) Parallel Channel	Provides an independent communication channel in hands-busy operational situations.
(5) Freedom of Movement	Within small physical regions speech can be used while moving about freely doing a task.
(6) Untrained Users	No training in basic physical skill required for use (as opposed to acquisition of typing or keying skills); speech is natural for users at all general skill levels (clerical to executive).
(7) Unplanned Communication	Speech is to be used immediately by users to communicate unplanned information, in a way not true of manual input.
(8) Identification of Speaker	Speakers are recognizable by their voice characteristics.
(9) Long Term Reliability	Performance of speech reception and processing tasks which require monotonous vigilant operation can be done more reliably by computer than by humans.
(10) Low Cost Operation	Speech can provide cost savings where it eliminates substantial numbers of people.

Fig. 4. Task demands providing comparative advantages for speech. (From Newell *et al.* [109].)

appear in a given context, or to reject a guess. When one of these systems makes errors, it is usually because the present state of its knowledge is incomplete and possibly inaccurate. In Section VI, we shall review aspects of system organization such as control strategies, error handling, real-time system design, and knowledge acquisition.

B. The Uses of Speech Recognition

Until recently there has been little experience in the use of speech recognition systems in real applications. Most of the systems developed in the 1960's were laboratory systems, which were expensive and had an unacceptable error rate for real life situations. Recently, however, there have been commercially available systems for isolated word recognition, costing from $10 000 to $100 000, with less than 1-percent error rate in noisy environments. The paper by Martin in this issue illustrates a variety of applications where these systems have been found to be useful and cost-effective.

As long as speech recognition systems continue to cost around $10 000 to $100 000, the range of applications for which they will be used will be limited. As the research under way at present comes to fruition over the next few years, and as connected speech recognition systems costing under $10 000 begin to become available, one can expect a significant increase in the number of applications. Fig. 4, adapted from Newell *et al.* [109], summarizes and extends the views expressed by several authors earlier [63], [78], [87], and [89] on the desirability and usefulness of speech—it provides a list of task situation characteristics that are likely to benefit from speech input. Beek *et al.* [17] provide an assessment of the potential military applications of automatic speech recognition.

As computers get cheaper and more powerful, it is estimated that 60–80 percent of the cost of running a business computer installation will be spent on data collection, preparation, and entry (unpublished proprietary studies; should be considered

speculative for the present). Given speech recognition systems that are flexible enough to change speakers or task definitions with a few days of effort, speech will begin to be used as an alternate medium of input to computers. Speech is likely to be used not so much for program entry, but rather primarily in data entry situations [33]. This increased usage should in turn lead to increased versatility and reduced cost in speech input systems.

There was some earlier skepticism as to whether speech input was necessary or even desirable as an input medium for computers [116]. The present attitude among the researchers in the field appears to be just the opposite, i.e., if speech input systems of reasonable cost and reliability were available, they would be the preferred mode of communication even though the relative cost is higher than other types of input [109]. Recent human factors studies in cooperative problem solving [23], [110] seem to support the view that speech is *the* preferred mode of communication. If it is indeed preferred, it seems safe to assume that the user would be willing to pay somewhat higher prices to be able to talk to computers. This prospect of being able to talk to computers is what drives the field, not just the development of a few systems for highly specialized applications.

II. Systems

This section provides an overview of the structure of different types of speech recognition systems. To accomplish this, one needs to answer questions such as: what are the important concepts and principles associated with each of these systems, what are their distinguishing features, how well do they perform, and so on. It is not always possible to answer these questions. Very few comparative results based on common test data are available. In many cases all the returns are not yet in. There are so many possible design choices that most systems are not strictly comparable with each other. Therefore, it will be necessary to restrict our discussion to somewhat superficial comparisons based on accuracy, response time, size of vocabulary, etc.

In this section, we will examine the structure and performance of the first three classes of systems shown in Fig. 1: *isolated word recognition systems, restricted connected speech recognition systems,* and *restricted speech understanding systems*. We will illustrate the principles of design and performance by picking a few systems which are representative of the state of the art in each category. For the sake of brevity, we will leave out the words "isolated" and "restricted" for the rest of this paper. Unless otherwise indicated, it is to be assumed that we are always talking about isolated word recognition systems, restricted connected speech recognition systems, and restricted speech understanding systems.

A. Word Recognition Systems (WRS)

Here we will look at the structure and performance of three systems by Itakura [70], Martin [96], and White [161]. Given a known vocabulary (of about 30 to 200 words) and a known speaker, these systems can recognize a word spoken in isolation with accuracies around 99 percent. The vocabulary and/or speaker can be changed but this usually requires a training session. These systems, though similar in some respects, have several interesting and distinguishing features. Fig. 5 summarizes some of the features of these systems that affect cost and performance. Researchers desirous of working in the field of word recognition would also benefit from studying the structure and features of several earlier (and somewhat lower performance) systems by Gold [51], Shearme and Leach [141], Bobrow and Klatt [18], Vicens [152], Medress [98], Valichiko and Zagoruiko [151], Vysotsky *et al.* [153], Pols [117], Von Keller [154], Itahashi, Makino, and Kido [67], and Sambur and Rabiner [135].

All three systems use the classical pattern recognition paradigm as their recognition strategy. The general paradigm involves comparing the parameter or feature representation of the incoming utterance with the prototype reference patterns of each of the words in the vocabulary. Fig. 6 presents the flow chart of a typical word recognition system. The main decisions to be made in the design of a word recognition system are: how to normalize for variations in speech; what is the parametric representation; how does the system adapt to a new speaker or new vocabulary; how does one measure the similarity of two utterances; and how to speed up the matching process.

Normalization: Even when the speaker and the microphone are not changed, variations in speech occur as a result of free variation from trial to trial, as well as the emotional state of the speaker and the ambient noise level of the environment. These result in changes in amplitude, duration, and signal-to-noise ratio for a given utterance. Before a match with stored templates can take place, some form of normalization is necessary to minimize the variability. Itakura [70] uses a second-order inverse filter based on the entire utterance to achieve noise and amplitude normalization. Martin [96] identifies several types of noise related problems: room noise, breath noise, intraword stop gaps, and operator-originated babble. Some of these result in incorrect detection of beginning and end of the utterance. Most of the noise problems can be overcome by careful attention to detail, such as close-speaking microphones, looking for spectra that are typical of breath noise, rejecting utterances that do not get a close match to any of the words, and so on. White, before measuring distances, normalizes all filter samples by dividing by the total energy.

Parametric Representations: Itakura uses linear predictive coding (LPC) coefficients, Martin uses hardware feature detectors based on bandpass filters, while White uses a 1/3-octave filter bank (see Section III). White [161] has studied the effect of using different parametric representations. Results of this experiment are given in Fig. 7. It shows that the 1/3-octave filter bank and LPC yield about similar results, and using a 6-channel-octave filter bank increases the error rate from 2 to 4 percent while doubling the speed of recognition. Transforming the parametric data into a pseudophonemic label prior to match can lead to significant reduction of storage but the error rate increases sharply to 9 percent. Reference pattern storage requirement is also affected by the choice of parametric representation. Assuming an average duration of 600 ms per word, White requires from 2160 to 7200 bits of storage (depending on parametric representation) per reference pattern and Itakura requires 4480 bits, while Martin requires only 512 bits per pattern.

Training: Change of speaker or vocabulary is accomplished in all three systems by training the system to produce a new set of reference patterns. Both Itakura and White use a single reference pattern per word. A single reference pattern cannot capture all variations in pronunciations for even a single speaker. Thus when a word exhibits higher than acceptable error rate it is desirable to store additional patterns. But this

	ITAKURA	MARTIN	WHITE
1. Transducer	Telephone	Close speaking microphone	Close speaking microphone
2. Noise level	68 dB (A)	90 dB (A)	65 dB (A)
3. Parametric representation	LPC	Hardware feature extractor	1/3 octave filter bank
4. No. of templates per word	Single template	Average of multiple templates	Single template
5. Space required per reference pattern	4480 bits	512 bits	7200 bits
6. Computer system	DDP-516	Nova, PDP/11 or Microcomputers	SIGMA-3

Fig. 5. Distinctive aspects of three word recognition systems. (Compiled from Itakura [70], Martin [95], [96], and White and Neely [161].)

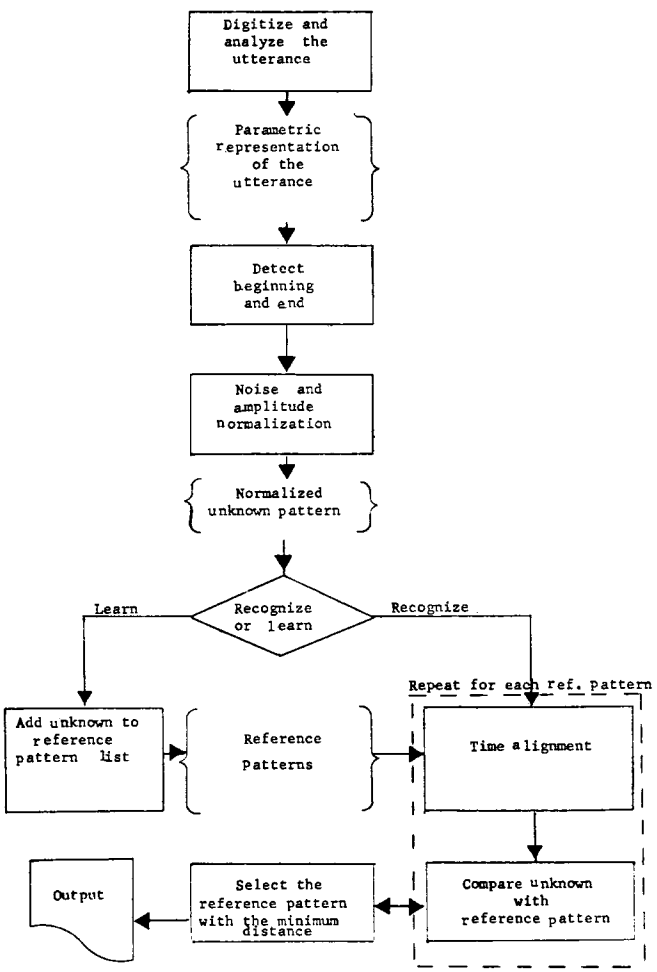

Fig. 6. Flow chart of a typical word recognition system.

Preprocessing method	Alpha-Digit vocabulary % correct	Recognition time per utterance	Data Rate bits per sec approximate
20 channel (1/3 octave filters)	98%	30 sec	12,000
LPC	97%	20 sec	4,200
6 channel (octave filters)	96%	15 sec	3,600
Phone code	91%	2 sec	500

Fig. 7. Effect of parametric representation on accuracy and response time of a system. Preprocessing produces four different parametric representations arranged in order of increasing data compression (lower bit rate). Recognition accuracy goes down as compression goes up. Phone code attempts to give a single pseudophonetic label for each 10-ms unit of speech.

requires additional storage and computation. Martin attempts abstraction of reference patterns by generating an average template from multiple samples.

Matching and Classification: Given an unknown input word, it is compared with every reference pattern to determine which pattern is most similar, i.e., has the minimum distance or maximum correlation to the unknown. This similarity measure is usually established by summing distances (or log probabilities as the case may be) between parameter vectors of the unknown and the reference. There are many design choices that affect the performance at this level, e.g., the choice of the basic time unit of sampling, the choice of the distance metric, differential weighting of parameters, and the choice of the time normalization function.

Itakura and White use dynamic programming for time normalization, while Martin divides the utterance into 16 equal time units. Itakura measures the distance between the unknown and the reference by summing the log probability based on residual prediction error every 15 ms. White measures the distance by summing the absolute values of the differences (Chebyshev norm) between the parameter vectors every 10 ms. Martin uses a weighted correlation metric to measure similarity every 40 ms or so (actually 1/16 of the duration of the utterance).

White shows that the nonlinear time warping based on dynamic programming is better than linear time scaling methods. He also shows Itakura's distance measure based on LPC linear prediction error yields about the same accuracy as other conventional methods. It is generally felt (based on speech bandwidth compression experiments) that significant loss of information results when speech is sampled at intervals exceeding 20 ms. However, note that Martin extracts averaged features based on longer time intervals and is not just sampling the signal parameters.

System	Vocabulary	Size	Noise	Microphone	Number of speakers	Accuracy (includes rejects if any)	Resp. time in times real time
Martin	Digits	10	—	Close speaking microphone (CSM)	10	99.79	Almost real-time
Martin	Aircraft ops.	11x12	—	CSM	10	99.32	Almost real-time
Martin	1 thru 34	34	90dB	CSM	12	98.5	Almost real-time
White	Alpha-digit	36	65	CSM	1	98.0	30
White	North Am. states	91	65	CSM	1	99.6	—
Itakura	Alpha-digit	36	68	Telephone	1	88.6	—
Itakura	Japanese geographical names	200	68	Telephone	1	98.95	22

Fig. 8. Performance characteristics of three word recognition systems. (Compiled from Itakura [70], Martin [95], [96], and White and Neely [161].)

Heuristics for Speedup: If a system compares the unknown with every one of the reference patterns in a brute-force manner, the response time increases linearly with the size of the vocabulary. Given the present speeds of minicomputers which can execute 0.2 to 0.5 million instructions per second (mips), the increase in response time is not noticeable for small vocabularies of 10 to 30. But when the size of vocabulary increases to a few hundred words it becomes essential to use techniques that reduce computation time. Itakura uses a sequential decision technique and rejects a reference pattern if its distance exceeds a variable threshold at any time during the match operation. This results in a speedup of the matching process by a factor of almost 10. White uses the duration, amplitude contour, and partial match distance of the first 200 ms as three independent measurements to eliminate the most unlikely candidates from the search list. Others have used gross segmental features [152] and pronouncing dictionary with phonological rules [67] in reducing search. But these require a more complex program organization.

Performance: Fig. 8 gives the published performance statistics for the three systems. It is important to remember that accuracy and response time are meaningful only when considered in the context of all the variables that affect the performance. Although recognition performance scores have been quoted only for systems ranging from 10 to 34 words, Martin's system has been used with vocabularies as high as 144 words. It is the only system that has been shown to work in very high noise (>90 dB) environments and with multiple speakers (using reference patterns which represent all the speakers). The accuracy of Itakura's system drops to 88.6 percent on the alpha-digit word list (aye, bee, cee, $\cdots$, zero, one, $\cdots$, nine). But note that it is the only system that uses a telephone as the transducer. In addition to restricting the frequency response to about 300 to 3000 Hz, the telephone introduces burst noise, distortion, echo, crosstalk, frequency translation, envelope delay, and clipping to list a few. In addition, the alpha-digit vocabulary is highly ambiguous. The fact that the system achieves about 1-percent error rate (and 1.65-percent rejection rate) on a less ambiguous 200-word vocabulary is indicative of its true power. White's system not only achieves high accuracies but also is notable for its system organization which permits it to use different parameters, different time normalization strategies, and different search reduction heuristics with ease. The important thing to remember is that each of these systems seems capable of working with less than 2-percent error rate in noisy environments given vocabularies in the range of 30 to 200. It seems reasonable to assume that accuracy will not degrade substantially with larger vocabularies. A useful indicator of this is the early system by Vicens [152] which achieved 91.4-percent with a 561-word vocabulary.

Future Directions: As long as the cost/performance requirements do not demand an order of magnitude improvement, the present systems approach will continue to be practical and viable. The improvements in computer technology have already brought the cost of such systems to around $10 000. However, if it becomes necessary to reduce the cost to the $1000 range, significant improvement to the basic algorithms will be necessary. The principal avenues for improvement are in the reference pattern representation and search strategies. Rather than storing a vector of parameters every 10 ms, it may be necessary to go to a segmentation and labeling scheme (see Section IV) as has been attempted by some earlier investigators [67], [152]. Rather than storing multiple reference patterns for multiple speakers, it will be necessary to find techniques for abstraction. It may also be necessary to use mixed search strategies in which a simpler parametric representation is used to eliminate unlikely candidates before using a more expensive matching technique. Since many of these techniques are essential for connected speech recognition, it is reasonable to assume that progress in that area will gradually lead to low-cost/high-performance word recognition systems.

B. Connected Speech Recognition (CSR)

In this section we will look at the structure and performance of four different connected speech recognition (CSR) systems: Hearsay-I and Dragon developed at Carnegie-Mellon University [7], [123]; the Lincoln system developed at M.I.T. Lincoln Laboratories [47], [48], [56], [97], [159], [162]; and the IBM system developed at IBM, T. J. Watson Research Center [10], [30], [31], [71], [72], [149], [150], [172], [173]. Hearsay-I was actually designed as a speech understanding sys-

tem, but the semantic and task modules can be deactivated so as to permit it to run like a connected speech recognition system. Both the Dragon and Lincoln systems were designed to add task-specific constraints later, but in their present form can be looked upon as connected speech recognition systems. These systems have achieved from 55- to 97-percent word accuracies. Since a sentence is considered to be incorrect even if only one word in the utterance is incorrect, the sentence accuracies tend to be much lower (around 30 to 81 percent). With tuning and algorithm improvement currently in progress, some of these systems are expected to show significant improvement in accuracy. Researchers interested in CSR systems might also wish to look at the papers in [26], [28], [95], [120], [148], and [152].

Why Is Connected Speech Recognition Difficult? When isolated word recognition systems are getting over 99-percent accuracies, why is it that CSR systems are straining to get similar accuracy? The answers are not difficult to find. In connected speech it is difficult to determine where one word ends and another begins. In addition, acoustic characteristics of sounds and words exhibit much greater variability in connected speech, depending on the context, compared with words spoken in isolation.

Any attempt to extend the design philosophy of isolated word recognition systems and recognize the utterance as a whole becomes an exercise in futility. Note that even a 10-word vocabulary of digits requires the storage of 10-million reference patterns if one wanted to recognize all the possible 7-digit sequences. Some way must be found for the recognition of the whole by analysis of the parts. The technique needed becomes one of analysis and description rather than classification (moving away from pattern recognition paradigms toward hierarchical systems, i.e., systems in which component subparts are recognized and grouped together to form larger and larger units).

To analyze and describe a component part, i.e., a word within the sentence, one needs a description of what to expect when that word is spoken. Again, the reference pattern idea of word recognition systems becomes unsatisfactory. As the number of words in the vocabulary and the number of different contextual variations per word get large, the storage required to store all the reference pattern becomes enormous. For a 200-word vocabulary, such as the one used by Itakura [70], a CSR system might need 2000 reference patterns requiring about 8-million bits of memory, not to mention the time and labor associated with speaking them into the machine. What is needed is a more compact representation of the sound patterns of the words such as those used by linguists, i.e., representation of words as a sequence of phones, phonemes, or syllables. This change from signal space representation of the words to a symbol space representation requires segmenting the continuous speech signal into discrete acoustically invariant parts and labeling each segment with phonemic or feature labels. A phonemic dictionary of the words could then be used to match at a symbolic level and determine which word was spoken.

Since CSR systems do not have the advantage of word recognition systems, of knowing the beginning and ending of words, one usually proceeds left-to-right, thereby forcing at least the beginning to be specified prior to the match for a word. Given where the first (left-most) word of the utterance ends, one can begin matching for the second word from about that position. One must still find techniques for terminating the match when an optimal match is found.

However, the exact match cannot be quite determined until the ending context (the word that follows) is also known. For example, in the word sequence "some milk" all of the nasal /m/ might be matched with the end of "some" leaving only the "ilk" part for a subsequent match. This is a special case of the juncture problem (see Section IV). Techniques are needed which will back up somewhat when the word being matched indicates that it might be necessary in this context. (see also Section VIII of Jelinek [72].)

Finally, error and uncertainty in segmentation, labeling, and matching make it necessary that several alternative word matches be considered as alternative paths. If there were 5 words in an utterance and we considered 5 alternative paths after each word, we would have 3125 (5^5) word sequences, out of which we have to pick the one that is most plausible. Selection of the best word sequence requires a tree search algorithm and a carefully constructed similarity measure.

The preceding design choices are what make CSR systems substantially more complex than word recognition systems. We do not yet have good signal-to-symbol transformation techniques nor do we fully understand how to do word matching performance of CSR systems when compared with word recognition systems. However, researchers have been working seriously on CSR techniques only for the past few years, and significant improvements can be expected in the not too distant future. The following discussion reviews the design choices made by each of the four systems (Fig. 9).

Front End Processing: The purpose of the front end in a CSR system is to process the signal and transform it to a symbol string so that matching can take place. The first three design choices in Fig. 9 affect the nature of this signal-to-symbol transformation. The Dragon system uses the simplest front end of all the systems. It uses the 10-ms speech segment as a basic unit and attempts matching at that level. Given a vector of 12 amplitude and zero-crossing parameters every 10 ms, the system computes the probabilities for each of 33 possible phonemic symbols. To account for allophonic variations, it uses multiple reference patterns (vectors) to represent each phonemic symbol.

Hearsay-I uses amplitude and zero-crossing parameters to obtain a multilevel segmentation into syllable-size units and phoneme-size units. Every 10-ms unit is given a phonemic label based on a nearest neighbor classification using a predefined set of cluster centers. Contiguous 10-ms segments with the same label are grouped together to form a phoneme-size segment. A syllable-like segmentation is derived based on local maxima and minima in the overall amplitude function of the utterance. These larger segments are given gross feature labels such as silence, fricative, and voiced.

The Lincoln system is described in detail by Weinstein *et al.* [159]. The fast digital processor (FDP) computes LPC spectra, tracks formant frequencies [97], performs a preliminary segmentation, and labels the segments as one of vowel, dip (intervocalic voiced consonants characterized by a dip in amplitude), fricative, and stop categories. Formant frequencies, formant motions, formant amplitude, and other spectral measurements are used in further classifying the segments into phone-like acoustic-phonetic elements (APEL) labels.

The IBM system front end is based on the approach developed by Dixon and Silverman [31], [32], for pattern

	Hearsay-I	Dragon	Lincoln	IBM
Parametric representation	Amplitude + zero crossings in 5 octave bands	Amplitude and zero crossings in 5 octave bands	LPC Spectra formants	Spectrum
Segmentation	Heuristic multilevel (syllabic + phonetic)	None	Heuristic	Heuristic
Labeling	Two level prot. matching	Prototype matching	Feature based	Prototype matching
Word matching	Heuristic	Stochastic	Heuristic	Stochastic
Phonological rules	Ad Hoc	None (can be added)	Yes	Yes
Word representation	Phonemic base form	Network	Phonemic base form	Network
Syntax	Productions anti-productions	Finite state network	Productions anti-productions	Finite state network
Search	Left to right search best first	Left to right search all paths	Left to right search best first	Left to right search using sequential decoding (similar to best first)

Fig. 9. Design choices of the four connected speech recognition (CSR) systems. (Compiled from Reddy *et al.* [123], Baker [8], Forgie *et al.* [48], Baker and Bahl [10], and other related publications.)

recognition using complex decision-making rules and dynamic segmentation [148]. The segmentation and labeling procedure uses energy, spectra, spectral change, an ordered list of five "most similar" classes, and their similarity values. The labeling is done by prototype matching as in the case of Hearsay-I and Dragon but using about 62 label classes.

Knowledge Representation: There are three types of knowledge that are usually required in a CSR system: phonological rules, lexicon, and syntax. The Dragon system has the most elegant representation of knowledge of the four systems [7]. All the knowledge is represented as a unified finite-state network representing a hierarchy of probabilistic functions of the Markov processes.

Hearsay-I organizes knowledge into independent and cooperating knowledge processes, which makes it easy to add or remove a knowledge source. The representation of knowledge within each process is somewhat arbitrary and depends on the needs of that process. Syntax is represented as a set of productions (generative rewriting rules) and antiproductions (analytic prediction rules). The lexicon contains only the phonemic base forms. Phonological information is embedded in various acoustic analysis procedures.

In the Lincoln system, syntactic constraints are represented by a set of production rules. Phonological and front-end-dependent rules are used to construct a lexicon from a base form dictionary [55], [56]. Other such rules are also applied during a heuristic word and matching process.

The IBM system uses a finite-state grammar and a directed graph representation of each lexical element [24]. Phonological rules are compiled into the lexicon. To account for the rules that do involve word boundaries, the graphs have multiple starting nodes labeled with conditions that must be met by preceding or following words. An important component of the IBM system is the extensive use of statistical information to provide transition probabilities within the finite-state networks representing task-dependent information. (See also Jelinek [72].)

Although both the Dragon and IBM systems use network representations and stochastic matching, they differ in several respects. Dragon uses an integrated representation of all the

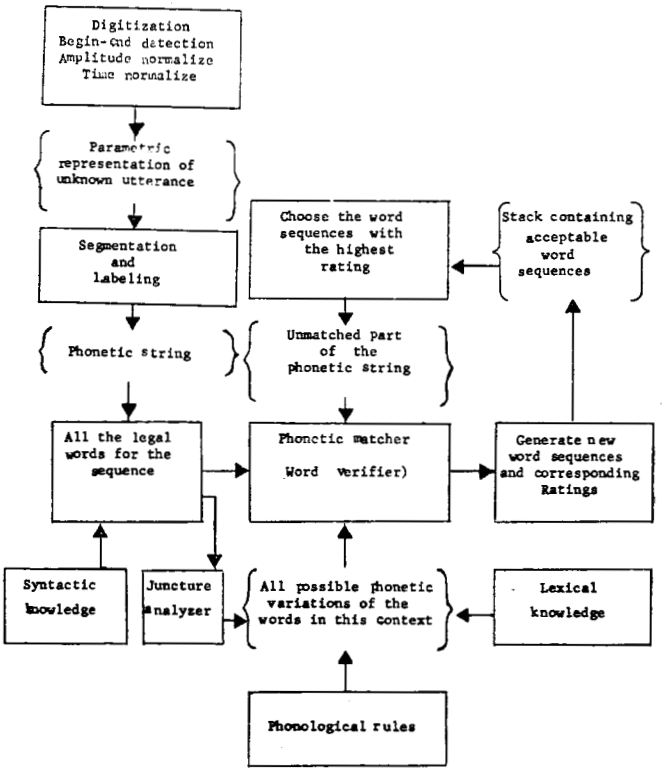

Fig. 10. Flow chart of a typical CSR system.

knowledge, whereas the IBM system has independent representations of the language, phonology, and acoustic components. Dragon evaluates the likelihood of all possible paths, while the IBM system uses sequential decoding to constrain the search to the most likely path.

Matching and Control: Fig. 10 is a flow chart of the recognition process of a typical CSR system. All the systems except Dragon use a stack (or a set) containing a list of alternative word sequences (or state sequences) arranged in descending order of their likelihoods (or scores) to represent the partial sentence analysis so far. Given the word sequence with the highest likelihood, the task-specific knowledge generates all the words that can follow that sequence. Each of these words is matched against the unmatched symbol (phonemic) string to estimate conditional likelihoods of occurrence. These are used to generate a new list of acceptable word sequences and their likelihoods. This process is repeated until the whole utterance is analyzed and an acceptable word sequence is determined. The Dragon system, rather than extending the best word sequence, extends all the sequences in parallel. The Markovian assumption permits it to collapse many alternative sequences into a single state, thus avoiding exponential growth.

The four systems differ significantly in the way in which insertion, deletion, and substitution errors are handled in the matching process, and the way in which likelihoods are estimated. Hearsay-I and Lincoln systems use heuristic techniques, while Dragon and IBM systems use the principles of stochastic modeling [72], [7] to estimate likelihoods. In Section IV, we will discuss techniques for word matching and verification in greater detail.

Performance: Fig. 11 gives some performance statistics for the four systems. The systems are not strictly comparable because of the number of variables involved. However, some

	Hearsay-I	Dragon	Lincoln	IBM
No. of Sentences	102	102	275	363
No. of Word tokens	578	578	-	-
No. of Speakers	4	1	6	1
No. of Tasks	5	5	1	2
Sentence Accuracy	31%	49%	49%	81%
Word Accuracy	55%	83%	-	97%
Response Time (x real-time)	9-44	48-174	15-25	25
Environment	Terminal room	Terminal room	Computer room	Sound booth
Transducer	CSM and telephone	CSM	CSM	HQM
Size of Vocabulary	24-194	24-194	237	250
Live Input	Yes	No	Yes	No
Date Operational	1972	1974	1974	1975
Computer	PDP-10	PDP-10	TX-2/FDP	360/91 and 370/168
Average No. of instructions executed per second of speech in million	3-15	15-60	45-75	30

Fig. 11. Performance statistics for four CSR systems. (From sources given for Fig. 9.)

general comparisons can be made. The IBM system has the best performance of the four, but one should bear in mind the fact that most of their results to date are based on relatively noise-free high-quality data for a single speaker. It is also the only system being improved actively at present. This tuning of the system should lead to even higher accuracies.

Hearsay-I and Dragon were run on the same data sets to permit strict comparison. Dragon yields significantly higher accuracy, though it is slower by a factor of 4 to 5. Hearsay-I yields much higher accuracies on tasks and speakers with which it is carefully trained (see Fig. 15). It was tested on several speakers and several tasks. As the vocabulary increases, its relatively weaker acoustic-phonetic module tends to make more errors in the absence of careful tuning. It was one of the first systems to be built and still is one of the very few that can be demonstrated live.

The Dragon system performance demonstrates that simple and mathematically tractable CSR systems can be built without sacrificing accuracy. Although searching all possible alternative paths becomes unfeasible for very large vocabularies, for restricted tasks with a few hundred word vocabulary, Dragon with its simpler program structure represents an attractive alternative.

The Lincoln system is the only one of the four that works for several speakers without significant tuning for the speaker. The 49-percent sentence accuracy represents the composite accuracy for all the speakers taken together. It was also tested with a 411-percent word vocabulary, yielding about 28-percent sentence accuracy over the same set of six speakers.

Future Directions: How can CSR systems achieve significantly higher performance and cost under $20 000? Better search, better matching, and better segmentation and labeling are all essential if the systems are to achieve higher accuracies. The best-first search strategy used by Hearsay-I and other systems leads to termination of search when it exceeds a given time limit. When this happens, it is usually because errors in evaluation have led to a wrong part of the search space, and the system is exploring a large number of incorrect paths. In most systems, this accounts for 20–30 percent of the sentence errors.

Dragon does not have the problem of thrashing since it searches all the possible extensions of a word (state) sequence. An intermediate strategy in which several promising alternative paths are considered in parallel (best few without backtracking), rather than all or the best-first strategies of the present systems, seems desirable. Lowerre [90] has implemented one such strategy in the Harpy system currently under development at Carnegie-Mellon University and has reduced the computation requirement by about a factor of 5 over Dragon without any loss of accuracy. The number of alternative paths to be considered is usually a function of the goodness of the parametric representation (and accuracy of the segmental labels). Continued research into this class of systems should lead to the development of low-cost CSR.

Accuracies in word matching and verification approaching those of word recognition systems, i.e., greater than 99 percent, are essential for the success of CSR. Since words exhibit greater variability in connected speech, this becomes a much more difficult task. Klatt [75] proposes the use of analysis-by-synthesis techniques as the principal solution to this problem. Near-term solutions include learning the transition probabilities of a word network using training data, as is being done by IBM, or learning the lexical descriptions themselves from examples, as is being attempted at Carnegie-Mellon University. There has been very little work on comparative evaluation of segmentation and labeling schemes. Further studies are needed to determine which techniques work well, especially in environments representative of real life situations.

C. Speech Understanding Systems (SUS)

In this section, we will study approaches to speech understanding systems (SUS's) design by discussing three systems, viz., Hearsay-II [36], [86], SPEECHLIS [166], and VDMS [127], [156], currently being developed, respectively, at Carnegie-Mellon University, Bolt Beranek and Newman, and jointly by System Development Corporation and Stanford Research Institute. We cannot give performance statistics for these systems as they are not working well enough yet. However, at least one earlier system, Hearsay-I, illustrates the potential importance and usefulness of semantic and conversation-dependent knowledge. Experiments on this system show that 25–30-percent improvement in sentence accuracies (e.g., from about 52 to 80 percent on one task) were achieved using chess-dependent semantic knowledge in the voice-chess task. Researchers interested in other attempts at speech understanding systems should look at [13], [20], [35], [47], [64], [123], [134], [152], [157], and [160].

What Makes Speech Understanding Difficult? In addition to the problems of having to recognize connected speech, SUS's tend to have the additional requirement that they must do so even when the utterance is not quite grammatical or well formed, and in the presence of speech-like noise (e.g., babble, mumble, and cough). The requirement is somewhat relaxed

by the concession that what matters in the end is not the recognition of each and every word in the utterance but rather the intent of the message. The systems are also required to keep track of the context of the conversation so far and use it to resolve any ambiguities that might arise within the present sentence. Clearly, one can attempt to build CSR systems with all the preceding characteristics and yet not use any task-specific information. Here we will restrict ourselves to the apparent differences in approach between the CSR and the SU systems of the current generation.

How do the above requirements translate into specific problems to be solved? We still have the problem of determining when a word begins and ends in connected speech, and the problem of wide variability in the acoustic characteristics of the words. But the solutions adopted in CSR systems to solve these problems do not quite carry over to SUS. One can no longer proceed left-to-right in the analysis of an utterance because of the possibility of error or unknown babble in the middle of the utterance. Thus the useful technique of keeping an ordered list of word sequences which are extended to the right after each iteration has to be modified significantly.

Another design choice of CSR systems that leads to difficulties is the notion that there is a bottom-up acoustic analyzer (the *front end*) which generates a phonemic (or some such) symbol string, and a top-down language model (the *back end*) predicting possible word candidates at that choice point, which are then compared by a matching procedure. As the vocabularies get larger, often the roles have to be reversed. One cannot afford to match 1000 possible nouns just because the grammar predicts that the next word might be a noun. In such cases, the phonemic string may be used to generate plausible word hypotheses, while the language model is used to verify such hypotheses for compatibility and consistency. In general, one wants systems in which the role of knowledge sources is somewhat symmetric. They may be required to predict or verify depending on the context. The representations of knowledge required to perform these different roles will usually be different.

In CSR we have seen that at a given time one of several words might be possible given the acoustic evidence. This is what leads to the nondeterministic search, i.e., consideration and evaluation of an ordered list of alternate word sequences in the flow chart given in Fig. 10. This nondeterministic (and errorful) nature of decisions permeates all the levels of the speech decoding process, i.e., segmental, phonetic, phonemic, syllabic, word, phrase, and conceptual, and not just the word level. There is no such thing as error-free segmentation, error-free labels, and so on up the levels. This requires the representation of alternate sequences at all levels, not just the word level as in the case of CSR systems. Fig. 12 (from Reddy and Erman [122]) illustrates the consequences of this nondeterminism.

At the bottom of Fig. 12, we see the speech waveform for part of an utterance: '. . . all about . . . '. The "true" locations of phoneme and word boundaries are given below the waveform. In a recognition system, the choices of segment boundaries and labels to be associated with each of the segments are not as clear cut. (In fact, even getting trained phoneticians to agree on the "true" locations is often difficult.) A segmentation and labeling program might produce segment boundaries as indicated by the dotted lines connecting the waveform to the segment level. Given the segmental features, the phoneme represented by the first segment might be /aw/,

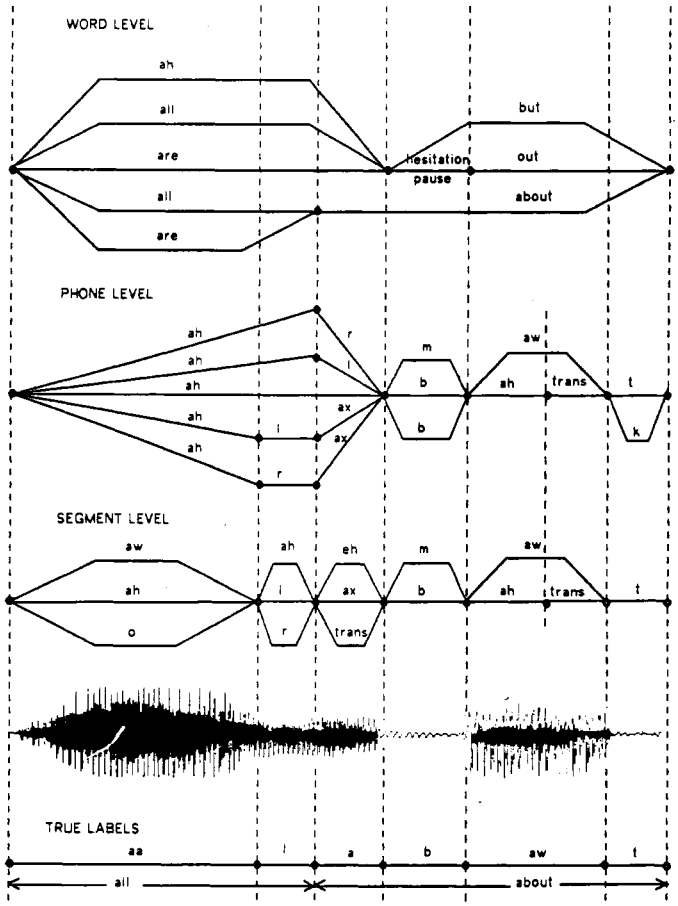

Fig. 12. Example of network representation of alternative choices at various levels. (From Reddy and Erman [122].)

/ah/, or /ow/. Similarly, several different labels can be given to each of the other segments. Given the necessary acoustic–phonetic rules, it is possible to combine, regroup, and delete segments, forming larger phoneme-size units, as shown in the figure. Note, for example, that /ah/ and /l/ are very similar, and it is not impossible that the minor parametric variability that caused the segment boundary at the lower level is just free variation. These phoneme hypotheses give rise to a multiplicity of word hypotheses such as 'ah but', 'all out', 'all about', 'all but', 'are about', and so on.

If, instead of selecting several alternate segmentations and following their consequences, we were to select a single segmentation and associate a single label with each segment, the resulting errors might make it impossible to verify and validate the correct word. Thus some form of network representation of alternate hypotheses at all levels is necessary in systems requiring high accuracy.

Even the lowest level decision about segmentation sometimes requires the active mediation of higher level knowledge such as the plausibility of a given word occurring in that position. Fig. 12 can be used to illustrate the point. The segment boundary at the word juncture of 'all' and 'about' is usually very difficult to find since the spectral characteristics of /l/ and the reduced vowel /ax/ tend to be very similar. In the event that a higher level process is fairly confident about this word sequence but there is no segment boundary, it could call upon the segmenter for a closer look, possibly using a different parametric representation. In general, SUS's require flexible means of cooperation and communication among different

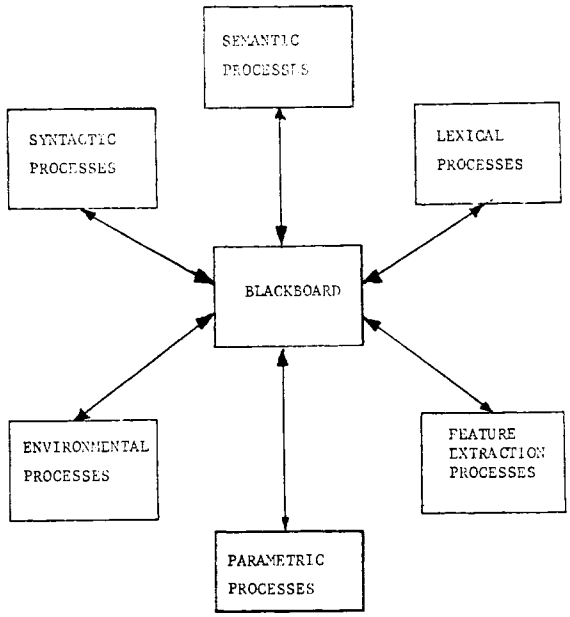

Fig. 13. Blackboard model used in Hearsay-II. (From Lesser *et al.* [86].)

knowledge sources (KS's). Since an SU system tends to have many more KS's than a CSR system, the system should be designed so that knowledge processes can easily be added and deleted.

Finally, the requirements of representation of *understanding, response generation, conversational context,* and *task* all add to the difficulty and overall complexity of an SUS.

Approaches to Speech Understanding: Given the difficulties that arise in SUS, it is clear that one needs significantly more sophisticated system design than those used in current CSR systems. At present, there is no clear agreement among researchers as to what an ideal system organization for an SUS might be.

In the VDMS system [127], the parser coordinates the operation of the system. In many respects the control flow resembles the one for CSR systems (Fig. 10) and is based on the best-first strategy. However, the simplistic notion of an ordered list of word sequences is replaced by a *parse net* mechanism which permits sharing of results and does not require strict left-to-right analysis of the utterance [115]. A language definition facility permits efficient internal representation of various KS's [128].

In the SPEECHLIS system [166], control strategies and system organization are derived through incremental simulation [168]. People located in different rooms simulate the various components and attempt to analyze an utterance by communicating via teletypewriter. Then one by one, people are replaced by computer algorithms having specified interface characteristics. A control strategy for SUS derived in this manner is described by Rovner *et al.* [131]. The final control structure is not available yet but is expected to be within the near future.

Perhaps the most ambitious of all the system organizations is the one used by the Hearsay-II system [86], [37]. Though it was designed with speech understanding systems in mind, it is viewed as one of the potential solutions to the problem of knowledge-based systems (KBS) architecture that is of general interest in artificial intelligence research. Other proposed solutions to the KBS architecture problem include Planner [62], production systems [106], and QA-4 [132].

Hearsay-II is based on a *blackboard* model (Fig. 13). The blackboard model conceives of each KS as an information gathering and dispensing process. When a KS generates a hypothesis about the utterance that might be useful for others, it broadcasts the hypothesis by writing it on the "blackboard"—a structurally uniform global data base. The hypothesis-and-test paradigm (see Section I-A) serves as the basic medium of communication among KS's. The way KS's communicate and cooperate with each other is to validate or reject each other's hypotheses. The KS's are treated uniformly by the system and are independent (i.e., anonymous to each other) and therefore relatively easy to modify and replace. The activation of a KS is data-driven, based on the occurrence of patterns on the blackboard which match the templates specified by the KS.

Most of the control difficulties associated with SUS appear to have a solution within the Hearsay framework. It is easy to delete, add, or replace KS's. The system can continue to function even in the absence of one or more of these KS's as long as there are some hypothesis generators and some verifiers in the aggregate. The blackboard consists of a uniform multilevel network (similar to the one in Fig. 12, but containing all the levels) and permits generation and linkage of alternate hypotheses at all the levels. A higher level KS can generate hypotheses at a lower level and vice versa. It is not necessary for the acoustic processing to be bottom-up and the language model to be top-down.

How does the recognition proceed in an asynchronously activated data-driven system such as Hearsay-II? Since there are not many systems of this type around, it is difficult for most people to visualize what happens. It is difficult to explain using flow charts which are primarily useful for explaining sequential flow of control. What we have here is an activity equivalent to a set of cooperating asynchronous parallel processors even when it runs on a uniprocessor. Generating and verifying hypotheses using several KS's is analogous to several persons attempting to solve a jigsaw puzzle with each person working on a different part of the puzzle but with each modifying his strategies based on the progress being made by the others.

What is important to realize is that within the Hearsay framework one can create the effects of a strictly bottom-up system, top-down system, or system which works one way at one time and the other way the next time, depending on cost and utility considerations. The ratings policy process, a global KS, combines and propagates ratings across levels facilitating focus of attention, goal-directed scheduling, and eventual recognition of the utterance. The focus-of-attention KS is used to determine an optimal set of alternative paths which should be explored further based on notions such as effort spent, desirability of further effort, important areas yet to be explored, and goal lists.

Knowledge Sources: Fig. 14 shows the design choices made by the three systems. Many of the low-level issues are common with CSR and do not require much discussion (see also Sections III, IV, and V). Here we will discuss the nature of the higher level knowledge sources used in each system.

The task for Hearsay-II is news retrieval, i.e., retrieval of daily wire-service news stories upon voice request by the user. The vocabulary size for the task is approximately 1200 words. The syntax for the task permits simple English-like sentences and uses the ACORN network representation developed by Hayes-Roth and Mostow [58]. The semantic and pragmatic model uses subselection mechanisms based on news items of

	Hearsay-II	Speechlis	VDMS
Microphone	Close speaking microphone	Close speaking microphone	Sony ECM-377 condenser microphone
Noise Level	Terminal room	Quiet office	Low (sound booth)
Parametric Rep.	LPC using Itakura metric	Formants and features	Formants and features
Segmentation	Parameter based	Feature based	Classification based
Labeling	Prototype matching	Heuristic	Heuristic
Word Hypothesis	Syllable based	Segment based	Syllable based
Word Verification	Markov Process	Analysis-by-Synthesis	Heuristic match with A-Matrix
Syntax	Restricted English	English-like	English-like
Semantics	Acorn net	Semantic net	Semantic net
Discourse Model	Topic based subselection	User model	Ellipsis and anaphora
Task	News retrieval	Travel Budget Manager	Submarine Data Base Management
Systems Control	Blackboard Model	Centralized Controller	Parser-Based

Fig. 14. Design choices of the three speech understanding systems. (Compiled from Lesser *et al.* [86], Woods [166], Ritea [127], and other related publications.)

1	2	3	4	5				6			
				Acoustic + Syntax				Acoustics + Syntax + Semantics			
Data set: Task/Speaker	Words in lexicon	No. of sentences	No. of words	%Sentences %Near miss	%Words	Average Time per sentence (sec)	Average Time per second of speech	%Sentences %Near miss	%Words	Average Time per sentence (sec)	Average Time per second of speech
1 Chess/Rn	31	14	82	43 100	87	11	6	100 100	100	9	5
2 Chess/Jb	31	19	86	74 95	93	12	9	100 100	100	8	6
3 Chess/Jb	31	21	105	15 50	69	15	8	48 90	88	13	7
4 Chess/(Tel) B1	31	25	99	52 84	78	7	6	80 88	88	7	6
Totals		79	352	46 80	81	11	7	79 93	93	9	6

Fig. 15. Performance of Hearsay-I speech understanding system. (From Erman [35].) Column 1 gives data set number, task, and speaker identification. Column 2 gives number of words in task lexicon. Column 3 shows number of sentences in data set. Column 4 gives total number of word tokens in data set. Column 5 gives results for HS-I system recognition with Acoustics module and Syntax module both operating. First subcolumn indicates percent of sentences recognized completely correctly. "Near miss" (indicated below that number in first subcolumn) indicates percent of times that recognized utterance differed from actual utterance by at most one word of approximate similar phonetic structure. Second subcolumn gives percent of words recognized correctly. Mean computation times on PDP-10 computer (in seconds per sentence and in seconds per second of speech) are shown in subcolumns three and four. Column 6 shows results for recognition using all three sources of knowledge (for Chess task only): Acoustics, Syntax, and Semantics modules. Subcolumns are similar to those of Column 5.

the data, analysis of the conversation, and the presence of certain content words in the blackboard.

The task for SPEECHLIS is to act as an assistant to a travel budget manager. It permits interactive query and manipulation of a travel budget of a company. It is meant to help a manager keep track of the trips taken or proposed, and to produce summary information such as the total money spent or allocated. The vocabulary is about 1000 words. The syntax permits a wide variety of English sentences and is based on the augmented transition network (ATN) formalism developed by Woods [164]. The parser is driven by a modified ATN grammar [15], [16] which permits parsing to start anywhere,

not necessarily left-to-right. The semantic component is based on a *semantic net* and uses *case frame tokens* to check for the consistency of completed syntactic constituents and the current semantic hypotheses [103]. Semantics is also used to focus attention and to produce a representation of the meaning of the utterance. A discourse model is used to predict what the user might say next.

The task for VDMS is to provide interactive query and retrieval of information from a "submarine data base." The vocabulary for the task is about 1000 words. The language definition facility [128] permits the speaker to communicate in relatively natural English. A semantic net representation along with strategies for net partitioning help to organize and condense the semantics of the task [60]. The pragmatic component [29] permits processing of simple forms of conversation-dependent ellipses and anaphora (see Section V-D).

Status: None of the three systems discussed here is working well enough to report performance statistics. Many component parts of these systems are working, but there are still weak links in the total systems that make such measurements meaningless. In the absence of these, it is useful to look at the performance of Hearsay-I to understand the potential role of semantics and other task-dependent knowledge in speech understanding systems. Fig. 15 shows performance results of Hearsay-I on the voice-chess task, both with and without the use of semantics. In a set of experiments using 79 utterances containing 352 words from three speakers (one speaker using telephone input), the sentence accuracies of the system increased from 46 to 79 percent by the use of task-dependent semantics. Response time also improved by about 10–20 percent (11–9 s). This speedup illustrates the fact that more complexity need not always mean slower systems. It is difficult to extrapolate how well other systems might perform from this lone example which used a powerful semantic module, i.e., a chess program, to predict legal moves. However, most researchers agree that without semantic and/or pragmatic knowledge, the systems will be unable to recognize or interpret some of the nongrammatical and non-well-formed sentences that usually occur in conversations.

Future Directions: It is too soon to draw any conclusions, but a few general observations can be made. The main scientific question to be answered in the next few years will be: is *understanding* a necessary prerequisite for *recognition* or can the simpler CSR systems do the job? If SU systems can achieve higher accuracies or recognize utterances faster on a given task than a CSR system, then the answer can be affirmative. It will always be the case for simple tasks that CSR systems, with their simpler program organization, are more likely to be adequate and cost-effective. What is not known are the tradeoffs that occur with large vocabulary English-like languages in complex task situations.

The second question is how to simplify the structure of the present SU systems while continuing to make effective use of all the available knowledge. Will SU systems price themselves out of the market by being more complex than most computer programs we write? The present systems are large, unwieldy, slow, and contain too many *ad hoc* decisions without the benefit of careful comparative evaluation of potential design choices. This is to be expected given their pioneering status. Once these systems are working, the most important task will be to study design alternatives that will significantly reduce the cost and/or increase the performance.

III. Signal Processing for Speech Recognition

One of the first decisions to be made in the design of a speech recognition system is how to digitize and represent speech in the computer. The tutorial-review paper, "Parametric Representations of Speech" by Schafer and Rabiner [137], provides a comprehensive and in-depth treatment of the digital techniques of speech analysis. Usually the first steps in the recognition process are the division of the connected speech into a sequence of utterances separated by pauses and the normalization of the signal to reduce variability due to noise and speaker. In this section, we shall review the signal processing techniques that have found to be useful in performing these tasks. The books by Fant [39] and Flanagan [44] are recommended for those interested in a thorough understanding of the theory and practice of speech analysis.

A. Parametric Analyses of Speech

Analysis of the speech signal was perhaps the single most popular subject of the papers on speech research in the 1950's and 1960's. The book by Flanagan [44] summarizes much of this work and provides an excellent description of the most commonly used speech analysis techniques, such as the short-time spectrum, formants, zero-crossings, and so on. In recent years, advances in computer architecture, graphics, and interactive use of systems have significantly altered the way speech research is conducted [105], and have tilted the balance in favor of digital rather than analog techniques. Here we will briefly mention some of the techniques that are commonly used in speech recognition research today.

The simplest digital representation of speech is pulse-code modulation (PCM). The changes in air pressure caused by the speech are sampled and digitized by a computer using an analog-to-digital converter. The speech is sampled from six to twenty thousand times a second in speech recognition research, depending on the frequency response desired. The speech is usually quantized at 9 to 16 bits per sample. Schafer and Rabiner [137] report that 11 bits is adequate for most purposes. The higher sample accuracy used in some systems [77], [175] provides for the wide dynamic range of signal levels observed across speakers. Conventional automatic gain control techniques produce signal distortions and are not preferred by researchers. It is easier to throw away the unneeded resolution later than to train speakers to speak at a uniform level of loudness.

Given a linear PCM representation of speech, one can derive other representations commonly used in speech recognition. Most of these representations are based on the assumption that parameters of speech remain unchanged over a short-time period. The commonly used *short-time spectrum* can be obtained using an analog filter bank or calculated from the PCM speech using digital filtering techniques or the fast Fourier transform (FFT).

The most widely used technique in speech recognition systems today is the LPC technique pioneered by Atal [1]–[5] and Itakura [68], [69], and extended by Markel [94] and Makhoul [91]. The tutorial-review papers by Makhoul [92], [93] discuss in detail the use of this technique in speech analysis and recognition. The theoretical foundations of LPC are based on the fact that the model of an acoustic tube whose input is impulses or low-level noise is mathematically tractable, i.e., the important parameters of the tube can be determined from its output, and that the model is good enough in estimating the response of the vocal tract. The basic ideal be-

hind LPC is that, given the acoustic tube model of the vocal tract, the present can be estimated from the immediate past, i.e., a speech sample can be expressed as a linear function of the preceding P speech samples. The coefficients of the linear function are determined by least square error fitting of the short-time speech signal. Given the coefficients, one can derive other parametric representations of speech, such as spectrum, formants, and vocal tract shape. Besides linear predictive coding (LPC), other commonly used parametric representations in speech recognition are the smoothed short-time spectrum based on DFT processing [175] and measurements based on amplitude and zerocrossings [11]. Silverman and Dixon [175] state that they have found DFT based processing of speech to be more desirable in their system than LPC based methods.

B. End-Point Detection

Division of connected discourse into utterances, i.e., detection of the beginning and ending of individual phrases is usually called end-point detection. Accurate determination of the end points is not very difficult if the signal-to-noise ratio is high, say greater than 60 dB. But most practical speech recognition systems must work with much lower S/N ratios, sometimes as low as 15 to 20 dB. Under these conditions, weak fricatives and low-amplitude voiced sounds occurring at the end points of the utterance become difficult to detect, leading to unacceptable error rates in high-accuracy systems. It is possible to detect end points accurately even in the presence of noise by careful attention to the algorithms. Gold [51], Vicens [152], and Martin [96] report on specific algorithms used by their systems.

The only careful study of this problem so far appears in a paper by Rabiner and Sambur [118]. They use the overall energy measure to locate an approximate end-point interval (N_1, N_2) such that although part of the utterance may be outside the interval, the actual end points are not within this interval. Precise end points are obtained by extending this interval in both directions to include any low-amplitude unvoiced fricatives at the beginning and end using a strict threshold on a zero-crossing measure.

Martin [96] reports on the use of a pattern matching technique to distinguish speech sounds from breath noise, which has its own spectral characteristics. Vicens [152] detects and rejects high-energy impulse noises, such as opening of the lips (when using a close-speaking microphone) and teletypewriter noise, by their very short duration followed by a long pause.

C. Noise Normalization

The single most important factor that affects the reliability and repeatability of speech recognition systems at present is the lack of proper attention to the sources of noise, i.e., background noise, microphone (or telephone) frequency response, reverberation noise, and noise from quantization and aliasing. Systems tend to be reliable as long as these sources of noise are kept invariant, but show a significant drop in performance as soon as the situation is altered. There have been a few studies on recognition in noisy environments [96], [104], and on the use of telephone as input device [38], [70]. But there are many questions that still remain unanswered.

Many system designs respond to the problems of noise by refusing to deal with this source of variability and requiring a new training session each time the environment, the microphone, or the speaker is changed. This seems acceptable in the short run but systematic studies into noise normalization techniques and the factors affecting the potential tradeoffs will be needed eventually.

Background Noise: This type of noise is usually produced by air-conditioning systems, fans, fluorescent lamps, typewriters, computer systems, background conversation, footsteps, traffic, opening and closing doors, and so on. The designers of a speech recognition system usually have little control over these in real-life environments. This type of noise is additive in nature and usually steady state except for impulse noise sources like typewriters. Depending on the environment, the noise levels will vary from about 60 dB (A) to 90 dB (A). Many of the noise sources have energies concentrated over certain portions of the spectrum and are generally not representable by white-noise experiments.

The most commonly used technique to minimize the effects of background noise is to use a head-mounted close-speaking microphone. When a speaker is producing speech at normal conversational levels, the average speech level increases by about 3 dB each time the microphone-to-speaker distance is reduced by an inch (when the distances are small). In absolute terms, the speech level is usually around 90 to 100 dB when the speaker-to-microphone distance is less than 1 in. Some systems use the so-called noise-canceling close-speaking microphones which are somewhat adaptive to the noise levels in the environment [126]. It is important that a close-speaking microphone be head mounted; otherwise even slight movement of the speaker relative to the microphone will cause large fluctuations in the speech level. Close-speaking microphones may also exhibit somewhat poorer frequency response characteristics. Careful experimental studies are needed to determine the microphone-related factors that affect the performance of a speech recognition system. Another method often used to reduce background noise derived from air conditioning, 60-Hz electrical hum, etc., is to high-pass filter the signal. (Cut-off frequency $\approx$80 to 120 Hz.)

Two signal processing techniques are generally used to normalize the spectra so as to reduce the effects of noise and distortion: inverse filtering and spectrum weighting. Silverman and Dixon [175] use quadratic spectral normalization to compensate for amplitude variations. Itakura [70] uses a second-order inverse filter based on the long-time spectrum of the entire utterance. This is intended to normalize the gross spectral distribution of the utterance. One could also use an inverse filter based on the noise characteristics alone rather than the whole utterance. Spectrum weighting techniques attempt to ignore those parts of the spectrum where the S/N ratio is low. This can be achieved by subtraction of the log noise spectrum from the log speech spectrum. This is equivalent to $[S_t + N_t]/N$, where S_t and N_t are the short-time spectra of speech and noise at time T, and N is the long-time average spectrum of the noise [170]. Another technique is to extract features only from those frequency bands where the signal spectrum exceeds the noise spectrum by at least 6 dB or more. The larger the difference, the less the impact due to noise.

Telephone Noise: Use of the telephone as the input device for a speech recognition system introduces several problems: the restriction of the bandwidth to 300 to 3000 Hz, the uneven frequency response of the carbon microphone, burst noise, distortion, echo, crosstalk, frequency translation, envelope delay, clipping, and so on. It is not known at this time how each of these problems affects the accuracy and performance of a system. In the case of isolated word recognition

systems, we know [70] that there is little effect in the case of relatively unambiguous vocabularies, but the accuracy drops significantly for ambiguous words, like in the alpha-digit list. Further studies are needed to suggest noise normalization techniques which will improve the accuracy for telephone input systems. Use of a different microphone headset connected to the telephone systems and preemphasis and/or postemphasis to normalize for the frequency response characteristics have been suggested [38]. Techniques suggested for background noise normalization would also be applicable here.

Reverberation Noise: In rooms which have hard reflecting surfaces, there is a significant reverberant field. Unlike background noise, which is additive, reverberation is a multiplicative noise and cannot easily be suppressed. If the reverberation measurement of an environment indicates a significant reverberation component, acoustic treatment of the room may be necessary. One can reduce the effects of reverberation by using a close-speaking microphone and locating the input station away from hard reflecting surfaces if possible.

Sampling Effects: Speech input is filtered by passing through a low-pass filter prior to the sampling process to eliminate undesired high frequencies (aliasing) of speech and high frequencies of noise. The characteristics of the filter, especially its rolloff near the cutoff frequency, are superimposed on the spectrum of the speech signal. It is not known whether this has any significant effect on the performance of the system.

Future Directions: We do not yet have a clear idea of which of the many possible techniques of noise normalization work well and which do not, and what the accuracy and performance tradeoffs are. We need many careful experimental studies. However, some basic issues are clear, although very few systems seem to have paid any attention to them. In high-noise environments, it will be difficult to detect, recognize, and distinguish between silence, weak fricatives (voiced and unvoiced), voice bars, voiced and unvoiced /h/, and nasals in some contexts. In systems using telephone input it will be difficult to detect and distinguish between most stops, fricatives, and nasals. Systems must provide for some form of noise adaptation at the symbol (usually phonemic) matching level if they are to be noise-insensitive.

IV. Task-Independent Knowledge

There are many aspects of processing that are common to both connected speech recognition (CSR) systems and speech understanding (SU) systems. The associated knowledge and techniques are: the speech sounds and symbols (phones and phonemes), features associated with speech sounds (acoustic-phonetics), rules governing the insertions and deletions of speech sounds (phonological rules), stress and intonation patterns of speech (prosodics), and matching of speech sounds with higher level linguistic units (syllables and words). The knowledge associated with these techniques is primarily dependent on the language and is usually independent of the task to be performed. In this section we will present the knowledge, techniques, and present state of accomplishment associated with these task-independent aspects of speech recognition, i.e., phonemic labeling, phonology, word hypothesis, and word verification.

In the following discussion, the use of the terms *phone*, *phoneme*, and *syllable* are somewhat different from the conventional usage of these terms in linguistic literature. In linguistics, the use of these terms is motivated by either perceptual or articulatory (production) considerations. Our use of the terms is acoustically motivated. Thus the minimally distinctive character of a phoneme is based on acoustic considerations, i.e., two phonemes are distinct if they are acoustically separable. We choose to use the same terms rather than invent some new ones because the intent is the same even through the criteria for distinguishabilty are different.

A. Phonemic Labeling

We have already seen, while discussing the difficulties of CSR systems (Section II-B), that some form of signal-to-symbol transformation is necessary and that attempting to match using reference patterns (as in the case of word recognition) can lead to unwieldy and unextensible CSR systems. The common technique adopted by CSR and SU systems is to attempt to transform the speech into a phonemic (or some such) string, which is then matched to the expected phones in the word. This transformation of the speech signal into a sequence of phonemic symbols usually involves feature detection, segmentation, and labeling. The tutorial-review paper by Broad and Shoup [21] presents some of the basic concepts associated with the phonemic labeling problem. The book by Fant [40] provides a comprehensive discussion of the features of speech sounds and their relationship to phonemic labels. Here we will outline the techniques useful in machine labeling of speech.

Feature detection usually represents the detection of silence, voicing, stress, and so on. The purpose of segmentation is to divide the continuous speech signal into discrete units based on some measure of acoustic similarity. Labeling schemes associate a phonemic symbol with each segmental unit. Before this symbol sequence can be used in matching, it is necessary to apply phonological rules (Section IV-B) to combine segments, change labels based on context, delete segments such as transitions, and so on.

Different systems do these operations in different orders. Some systems segment the data first, use the averaged segmental parameters to detect features such as voicing and stress, and then attempt labeling. Other systems classify and label speech every 10 ms and use the resulting labels in segmentation. The former tends to be less sensitive to noise and segment boundary effects. The commonly used paradigm is to detect features, then segment, and then label.

Not all connected speech recognition systems use segmentation and labeling prior to matching. The Dragon system [9] matches the phonemic representation of the word directly at the 10-ms level. Bakis [12] reports significantly improved performance on the continuous digit recognition experiment using the segmentation-free word template matching technique. However, such techniques are likely to run into serious difficulties with large vocabularies involving a wide variety of juncture phenomena.

1) Feature Extraction: There are a large number of potential features one can extract from speech [40]. Many of these tend to be unreliable, and it is difficult to devise detection algorithms for them. Hughes and Hemdal [65] and Itahashi *et al.* [67] have attempted detecting various distinctive features with a limited success. However, certain basic features are extracted by almost all the systems: silence, voicing, and stress. These are usually based on two measurements: energy and fundamental frequency (pitch period) of the speech signal. Schafer and Rabiner [137] present the basic concepts and techniques useful in extracting energy and pitch measure-

ments. One common measure of energy is to sum the absolute values of the speech samples over a 5-ms region. The papers by Sondhi [147], Gold and Rabiner [52], and Gillmann [50], provide several alternate techniques for extracting pitch periods.

Silence detection is usually based on a threshold applied to the energy function. Weinstein *et al.* [159] use a threshold of about 3 dB over the background noise level. When the energy falls below this threshold it is classified as a silence. The silence segment is then extended on both sides as long as the energy in the adjacent frames stays below a slightly higher threshold.

Voicing decision is usually based on the results of pitch extraction. If the period shows wide variability, it is treated as unvoiced [52]. In addition, concentration of energy in the high-frequency regions (3700–5000 Hz) when compared with low-frequency regions (100–900 Hz) is also an indication of unvoicing [159].

Stress decisions are usually based both on energy and on the pitch period. Much of the recent work on stressed syllable detection is based on the work by Lea *et al.* [79], [81]. Lea gives an algorithm for detecting stressed syllables from the rise–fall patterns of the pitch period (F_0) contours and the local maxima of energy. Although F_0 contours fall gradually in each breath group, it is possible to detect those maxima related to stress and they usually show rises over the gradually falling contour.

In addition to such primary features, several systems [136], [127] use other measures such as normalized linear prediction error, frequency of 2-pole linear prediction model, first autocorrelation coefficient, and energy in the 5–10-kHz region. Each of these features measures some quality of the signal such as lack of high-frequency energy or presence of voicing.

2) Segmentation: It is often said that "the problem with segmentation is that you can't segment." That is to say, there is no simple machine algorithm which will give phonemic boundaries. However, if one is satisfied with acoustic boundaries, i.e., boundaries associated with significant changes in the acoustic characteristics of speech, then it is possible to devise automatic algorithms to segment speech. The papers by Fant and Lindblom [42], Reddy [119], Reddy and Vicens [125], Tappert *et al.* [148], Dixon and Silverman [30], [31], Baker [11], and Goldberg [53] illustrate several attempts to segment connected speech into phonemic units with the understanding that some phonemic boundaries may be missed and some acoustic boundaries may be inserted where there are no phonemic boundaries.

Why is segmentation difficult? Figs. 16 and 17 (from Goldberg [53]) give examples where a boundary may be missing or added based on acoustic evidence. Fig. 16 gives the oscillogram of part of the word sequence 'been measured.' Note that it is impossible to say where /n/ ends and /m/ begins (vide the time period 121 to 132 centiseconds). The labels indicated at the bottom of the figure indicate an arbitrary choice on the part of the human segmenter. Fig. 17 contains part of the waveform of the word 'samples' and the corresponding spectrogram. Note that there is a significant variation in the parameters in the last 4 centiseconds (from 75 to 79) of the vowel resulting from coarticulation. Any attempt to ignore this variation can lead to errors in other contexts. A segmentation program based on acoustic measurements would normally insert an extra segment even though there is not a corresponding phoneme.

Fig. 16. Example of missing phoneme boundary, showing oscillogram (waveform) plot of part of the word sequence "been measured." Time scale is indicated on the first line below the plot in centisecond units. Second line shows a manually derived marker indicating which part of the waveform belongs to "been" and which part to "measured" (only parts of each word are visible in the plot). Third line shows manually derived markers indicating where various phonemes belonging to the words begin and end. The ending phoneme /n/ of the word "been" is assimilated with the beginning phoneme /m/ of the word "measured" as can be seen from the lack of any visible indication in the waveform. Boundary indicated at time 128 cs represents an arbitrary choice on the part of the human segmenter.

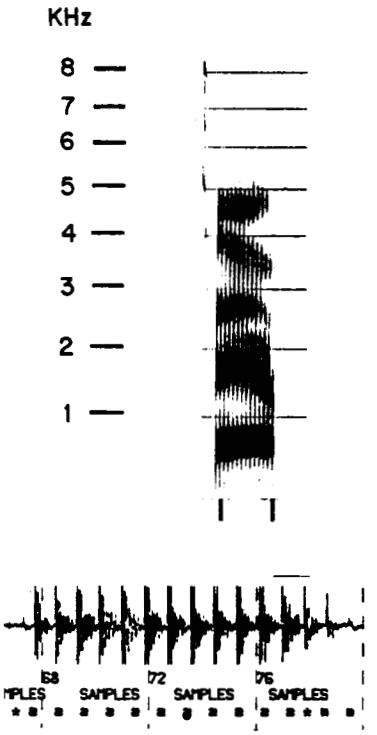

Fig. 17. Example indicating possibility of extra acoustic segments for a given phonetic segment. The waveform plot is similar to the one described in Fig. 16. The vowel /æ/ has significantly different parameters for the last 50 ms or so (from time unit 75 onwards) making it a candidate for an extra acoustic boundary.

There are many times when a boundary is expected but it is difficult to say exactly where it should be placed because of continuous variation. Fig. 18 illustrates an example of this. Looking at the waveform and the spectrogram for part of the word 'ratio,' it is difficult to say where /i/ ends and /o/ begins. Again the boundary indicated (* on line 3 between /i/ and /o/ at time 324) represents an arbitrary choice on the part of the human segmenter. If a machine segmentation program should place that boundary marker at a slightly different position (or time unit), it does not necessarily indicate an error of segmentation. In general, one will observe a few missing phonemic boundaries, some extra boundaries, and some cases in which the location of the boundary is shifted. These do not necessarily represent errors, but rather arise from the nature of the acoustic phenomena associated with phonemic symbols.

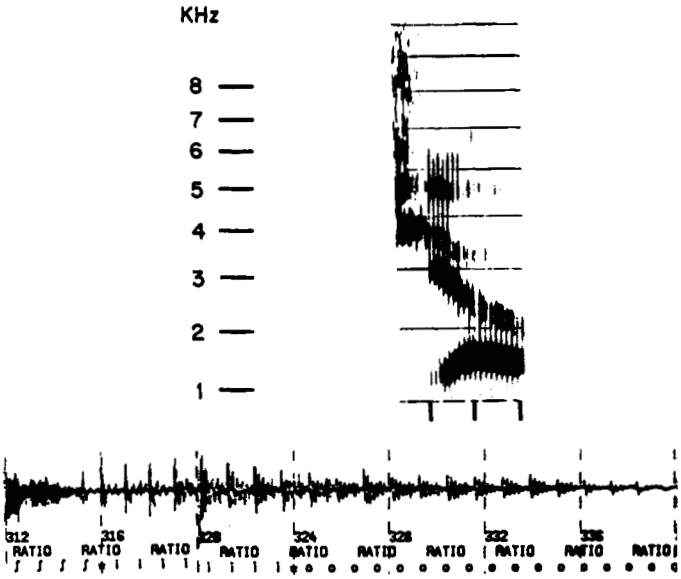

Fig. 18. Example of hard-to-place boundaries. The waveform plot is similar to the one described in Fig. 16. Note that it is difficult to say where /i/ ends and /o/ begins in the word "ratio"—boundary could be placed anywhere between time unit 324 and 330.

Another problem with segmentation is devising an appropriate acoustic similarity measure which indicates a boundary if and only if there is significant acoustic change. It is difficult to express in algorithm from intuitive notions of waveform similarity.

Segmentation techniques: Amplitude (or energy) is the single most important measure in segmentation [119]. It can be used to detect many of the boundaries. Sometimes it leads to extra segments in fricative sounds and missing segments in vowel/liquid sequences. Spectral characteristics of the sounds can be used to find additional boundaries that are missed by the gross segmentation based on amplitude. This additional segmentation may be based on heuristic techniques using speech specific information [31], [32], [136], [159], or algorithmic techniques based on a similarity metric on the parameter space [53]. The latter technique permits experimentation with and evaluation of different parametric representations of speech. The instantaneous frequency representation [11] yields the most accurate boundaries in nonsonorant speech (within 1 or 2 ms).

Fig. 19 shows typical segment boundaries placed by a machine segmenter [53]. The labels on the third and fourth rows under the waveforms in Fig. 19 show the human segmentation and labeling given for the utterance. The vertical lines indicate machine segmentation. Note that there are many more machine segments than there are phonemic boundaries. Examination of the waveform and spectrograms in the figure shows that many of the extra segments are in fact not errors in segmentation but a consequence of intraphone variability and transitions between phones.

Performance: There are many factors which must be considered in the evaluation of segmentation programs. Fig. 20 presents a list of some of the more important ones. Since reported performance evaluations of different programs may deal with only a few of these factors, direct comparisons are difficult to make.

Dixon and Silverman [31] report 6.9 percent missed segments with 10.5 percent extra segments over 6175 segments (8.5 min of speech). Recordings were made under sound booth conditions with high quality equipment. A single speaker was used and speaker specific training was employed. The segmenter also makes use of speech specific knowledge at the phonetic level. Dixon and Silverman report that recent work has further reduced these error rates to 5 percent missed and 6 percent extra.

Baker [11] reports 9.3 percent missed with 17.6 percent extra segments for 216 segments in 5 sentences spoken by 4 male and one female speaker. Recording conditions and equipment varied, and no speaker specific information at the phonetic level was used. The referent segmentation used in Baker's study included all segments indicated at the phonemic level. Hence, omissions of sounds by the speakers may have caused slightly higher missed segment rates than would be shown with a less conservative referent.

Goldberg [53] reports a 3.7 percent missed and 27.6 percent extra segments for 1085 segments in 40 sentences spoken under terminal room conditions (~65 dB (A)) using a close speaking microphone. Training was speaker specific and no speech specific knowledge was employed. The system was tested for several speakers with similar results. Goldberg introduces a model from signal detection theory to quantify the missed versus extra segment tradeoff. By adjusting a few thresholds, error rates (predicted by the model) of 11.7 percent missed and 11.7 percent extra can be achieved.

3) Labeling: Labeling is the process by which each segment is assigned a label. This can be done either heuristically [73], [120], [136], [159], based on speech specific information, or by using a nearest neighbor pattern classification technique [53], or both [31].

As in every other aspect of speech recognition, it is difficult to attempt comparative evaluation of different labeling schemes. There are three factors that affect accuracy and computation time of a labeling procedure: number of labels, number of alternate choices, and the correctness criteria. Weinstein *et al.* [159] report results of labeling accuracy for about 20 or the 35 APEL labels used in the Lincoln system. Dixon and Silverman [31] use 34 different phonemic class labels in their system. Goldberg [53] uses about 70 labels to account for various allophonic variations. The finer the desired phonemic transcription, the lower the accuracy.

The second factor that affects apparent accuracy of labeling is the *branching factor*, i.e., the number of alternate labels assigned to a segment, and is indicative of the degree of indecision of an algorithm. If you permit a large number of alternative choices, the correct label will appear sooner or later. Fig. 21 (from Goldberg [53]) illustrates the effects of the branching factor and the number of labels on accuracy. The figure shows that the labeling accuracy increases sharply when one considers several alternative choices. For example, the correct label appears as one of the choices 70–80 percent of the time if one considers 5 alternative choices (out of a possible 40) whereas it is the top choice only about 35 percent of the time.

The third and perhaps the most elusive factor is the label assigned to a segment by a human subject to be used for the evaluation of the machine labeler. Dixon *et al.* [174], [144] discuss the need for objective phonetic transcription and its importance in obtaining reliable performance statistics. Shockey and Reddy [142] show that subjective judgments of phoneticians seem to agree among themselves only about 51 percent of the time when labeling spontaneous connected speech in unfamiliar languages. Most of this variability is at-

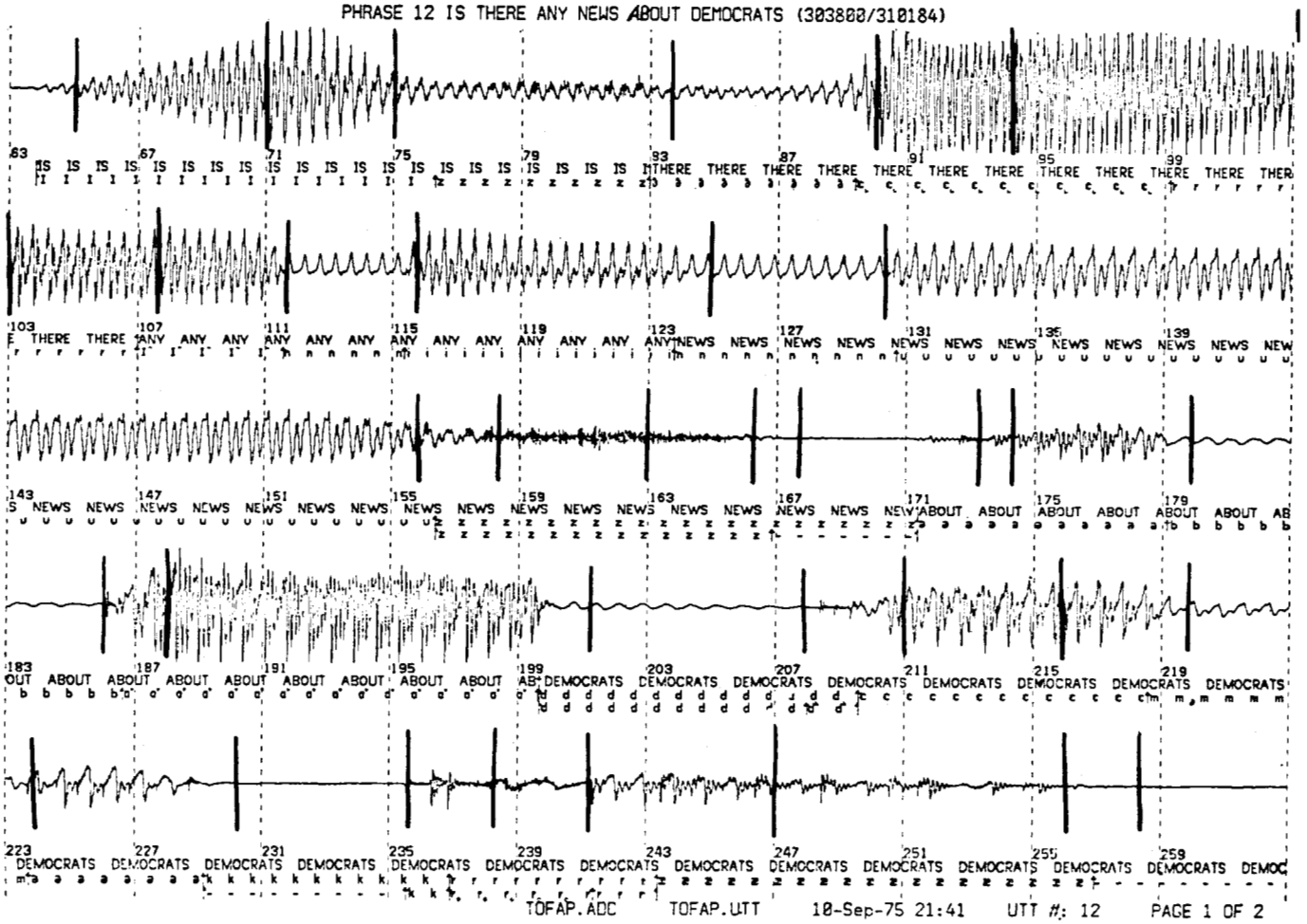

Fig. 19. Typical result of a machine segmentation "Is there any news about Democrats?" The waveform plot is similar to the one described in Fig. 16. Dark vertical lines indicate position of machine boundaries.

1.	Recording conditions	Environmental noise Recording equipment (microphone!) Noise and amplitude pre-processing
2.	Speaker variations	Number, sex, and age Speaker specific training
3.	Use of speech specific knowledge	Phonetic and coarticulation rules
4.	Performance measures	Missing vs. extra segment error rates Cost in speed and memory
5.	Confidence of results	Quantity of data Nature and quality of "correct" segmentation (referent)

Fig. 20. Typical factors which may affect segmentation performance evaluations.

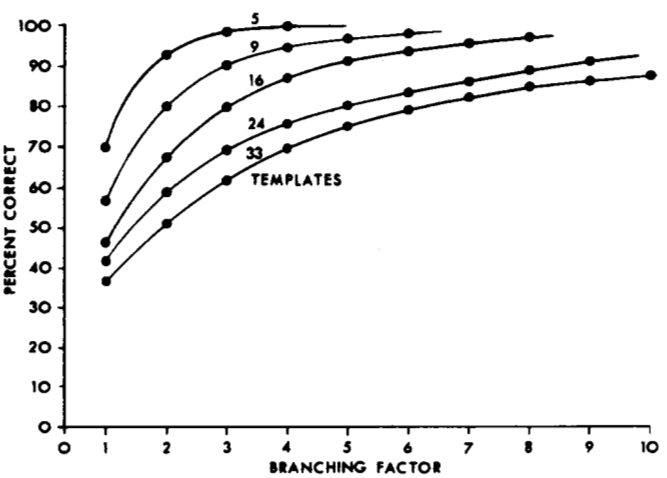

Fig. 21. Accuracy of labeling as a function of branching factor (number of choices per segment) and number of templates. The number below each curve indicates the number of template classes. The 5 classes consist of vowels, liquids, fricatives, nasals, and stops. Larger numbers of classes indicate finer subdivisions. The 33 classes are individual phonetic labels. (For further details, see Goldberg [53].)

tributable to the unfamiliarity with the language and the undistinctive character of many of the sounds in connected speech spoken by a native speaker. Carterette [22] says that phoneticians disagree on phonemic labels up to 20 percent of the time on a familiar language. Shoup [143] says that phoneticians with similar training and background disagree less than 10 percent of the time on phonetic labels while transcribing a familiar language. It is not known at this time how often the higher level linguistic cues are instrumental in determining the phone label and how often all the relevant information is available in the actual speech signal itself. Since we need some form of human segmentation and labeling to compare machine labeling with, it is clear that the accuracies reported are only as good as the manually derived labels we start with. Fig. 19 illustrates some of the problems with perceptual judgments associated with manually derived labels.

Some systems [159], [166] attempt *speaker normalization* based on vocal tract length estimates [155] or a vowel normalization procedure proposed by Gerstman [49]. The latter procedure observes the range of values for formant 1 and formant 2, and uses a linear scaling technique based on these values. Other systems [30], [53] achieve speaker normalization by using speaker-dependent prototype reference patterns associated with each label. Note that speaker normalization may be necessary and useful at many other levels besides the labeling level. Fig. 3 indicates many of the potential sources of improvement. In addition to vocal tract variations, one has to consider dialectal variations, vocabulary and grammatical preferences of the user, and a psychological model of the user predicting what action he might take next.

Performance: There are a few performance evaluations of labeling schemes given in the literature [31], [46], [53], [120], [159]. Of these, the performance results given by Dixon and Silverman [31] are the most comprehensive so far and are representative of the state of accomplishment to date in labeling connected speech. Dixon and Silverman report 61.7 percent accuracy at the phoneme level and 88.6 percent accuracy at the phoneme-class level.

B. Phonological Rules

In the preceding section, we saw how a speech utterance is segmented and each segment is labeled with one (or more) of a set of phonemic symbols. Here we will give some examples of knowledge and rules that have been found to be useful in combining and relabeling segmental units (with essentially phonemic labels). In natural continuous speech, the influence of surrounding vowels and consonants and stress patterns can lead to insertions and deletions of segments or variation in the expected acoustic characteristics of phonemes. The rules that govern this behavior are called *phonological rules.* Papers by Cohen and Mercer [24] and Oshika *et al.* [112] provide many examples of the nature and use of phonological rules in speech recognition systems.

Fig. 12 illustrates the need for rules that combine adjacent segments into larger sized units. In the word "about" the diphthong /aw/ is usually divided into two or more segments. Given the labels (or features) of individual segments, one can define diphthong detection rules which will combine the segments into larger units when an appropriate sequence of labels occurs. Often the onglide and offglide portion of a vowel are indicated as separate segments. In this case these segments are deleted using segment deletion rules. In Fig. 12, the initial part of /aa/ in "all" is an example of an onglide which has no phonemic significance of its own.

Fig. 22 gives some examples of insertion, deletion, and change rules that are used in CSR and SU systems. The comments associated with each rule explain the applicability of that rule. It is estimated that a few hundred such rules may be needed to explain the commonly occurring phonological variations.

Segment insertion rules are sometimes used to propose extra phonemic boundaries where no acoustic boundaries exist. Fig. 16 shows an example of word juncture where the ending sound of one word is the same as (or similar to) the beginning segment of the following word. This usually leads to a single longer duration acoustic segment. If the duration exceeds a threshold for that class of speech sounds, an extra boundary may be inserted (usually at the midpoint) to facilitate word matching.

Fig. 23 (from Oshika *et al.* [112]) illustrates the phonological variation that is pervasive in natural continuous speech. Notice the significant changes in the formant trajectories as one goes from isolated words to connected speech. It is especially noticeable in the realization of the words "you" and "the" in the connected utterance. In both these cases, the expected vowel characteristics have been significantly altered because of the context and stress. Any attempt to find a /u/-like sound in "you" during the word matching would lead to the rejection of that word as a possible choice for that portion of the utterance.

Fig. 23 also illustrates several other phonological phenomena. Note that a frication segment is inserted at the juncture of the words "did you" (pronounced as "did ja") where there was none in the isolated words. The fricative sounds at the juncture of "refresh screen" are merged into a single /sh/ sound, leading to the deletion of a segment. Oshika *et al.* [112] and Cohen and Mercer [24] provide a detailed discussion of the specification and use of phonological rules in speech recognition systems.

C. Prosodics

Prosodic features of speech, i.e., stress, intonation, rhythm, pauses, and tempo, augment the syntactic and semantic structure of language in helping to communicate the intended message. *Stress* patterns of speech help to distinguish between "light housekeeper" and "lighthouse keeper." *Intonation* helps to distinguish between "I will move, on Saturday" and "I will move on, Saturday." *Rhythm* in speech is illustrated by the example "John, who was the best boy in school, got the medal" where one usually observes an increase in the speech rate during the production of the parenthetical relative clause (almost as though each constituent of the sentence has to observe an equal-time rule). *Tempo* of speech and *pauses* provide additional distinctive patterns helpful in the interpretation of spoken language. These and other examples of prosodic knowledge, given in Lea *et al.* [81], illustrate the importance of prosodics in speech understanding research.

Stress and pause structure have been used to determine syntactic boundaries in utterances [81], [82], [111]. Although the boundaries cannot be placed exactly, they do indicate the general area within the utterance where a syntactic boundary may be expected. Rising and falling patterns of pitch contours have been used to determine whether an utterance is a question or an assertion. Research is presently under way [81] to determine other acoustic correlates of contrasting patterns of intonation, rhythm, and tempo in speech.

X → W/Y * Z means X can become W in the context of Y and Z

{ } means logical OR

( ) means enclosed segment is optional

;... indicates a comment

Insertion Rules

0 → (~) / n * s	; a silence segment may be inserted between /n/ and /s/
r → (s) (r) / t * i	; an /r/ occuring in the context of /t/ and /i/ may result in an optional fricative sound /s/
0 → (i) / t * u	; an /i/-like segment may appear in the context of /t/ and /u/

Deletion Rules

b → 0 / m *	; a /b/ might be missing when preceded by an /m/
{p, t, k} → 0 / * {p, t, k}	; an unvoiced stop may be missing in the context of another unvoiced stop

Substitution Rules

z → s / Unvoiced * Unvoiced	; a /z/ becomes devoiced in an unvoiced context
s → z / Voiced * Voiced	; an /s/ becomes voiced in a voiced context
{t, d} → Flap / Vowel $_1$ * Vowel $_3$	; an intervocalic /t/ or /d/ become flap-like

Fig. 22. Some typical phonological rules. (Compiled from Cohen and Mercer [24], Erman [35], and Hall [55].)

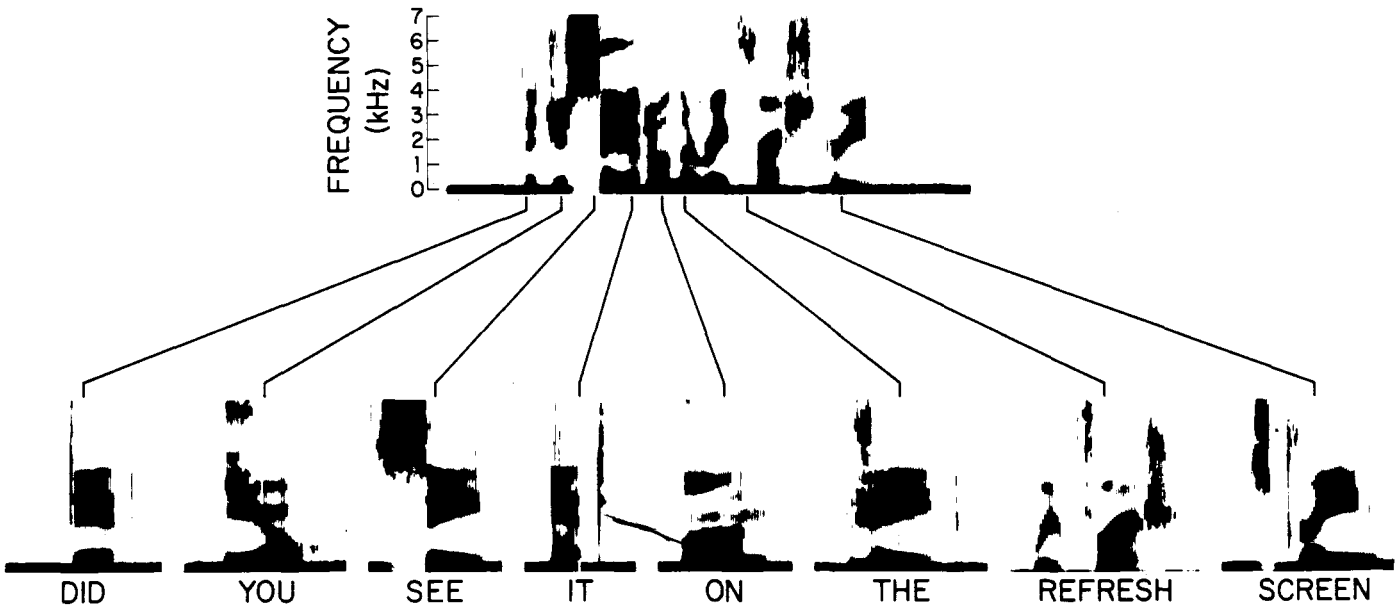

Fig. 23. Spectrograms of the words "did you see it on the refresh screen?" said in isolation, compared with spectrogram of the words in the continuous phrase. Note the significant differences in acoustic characteristics of the words "you" and "the" in the connected speech compared to the corresponding isolated words. (From Oshika *et al.* [112].)

Future Directions: It has been said that "prosodics play the role in spoken language that 'space' plays in written language." If this is so, we have been slow in making effective use of this source of knowledge. To be sure, amplitude (which is a measure of stress) has been used in segmentation and stress detection for a long time, but we are only just beginning to explore other uses of stress and intonation-related phenomena. Forgie [45] suggests that much of the nonlinguistic phenomena such as hesitation and stutter may be detectable from prosodic patterns of speech. This would make it easier to detect and ignore some of the speech-like noises and other nongrammaticality in spoken utterances. These considerations make prosodics an important area of study in speech recognition research.

D. Word Hypothesis

As the vocabularies get larger, it becomes expensive to match all the possible words that may appear in a given part of the

utterance. In large vocabulary systems, the phonemic string is used to generate hypotheses for plausible words in a given location which are then verified by various higher level processes and/or more expensive low-level verifiers. There are two techniques presently being used in SU systems [100], [130], [146].

Rovner *et al.* [130] use partial sequences of phonemic descriptions and the expected phonemic patterns of the vocabulary to retrieve words from the lexicon that match the acoustic characteristics of the signal to some specified degree. The class of words scanned may be delimited by explicit enumeration, class membership, or phonemic length. Appropriate word boundary rules are used whenever an adjacent word is known.

Smith's word hypothesizer [146] detects syllables and uses them to retrieve and verify words that have matching syllable types. The syllable type is hypothesized using a Markov probability model to relate a sequence of phones to a sequence of states defining a syllable type. For each stressed syllable type, the program looks up all the words containing the same syllable type using an inverted lexicon, prunes away multisyllable words that do not match with adjacent syllable hypotheses, and rates and hypothesizes the remaining words. Smith reports that the program locates the correct word within the first two choices about 59 percent of the time given a 275-word vocabulary.

E. Word Verification

Matching and verification of hypothesized words, given the acoustic evidence from an unknown utterance, is basic to almost all speech recognition systems. In connected speech this usually implies matching the expected phonemic realizations of a given word with an unknown phonemic string possibly containing insertion, deletion, and substitution errors. Here we will illustrate the techniques employed by looking at the structure of three word verification techniques: heuristic matching [35], [55], stochastic matching [71], [150], [172], and analysis-by-synthesis [75], [12].

Heuristic Matching: Many of the connected speech recognition systems use this type of matching. Hall [55] of Lincoln Laboratories gives one of the clearest descriptions of the basic techniques involved. Matching process must account for three types of errors: some of the symbols in the phonemic string may be spurious (*insertion error*); some of the expected phonemes may be missing as a result of incorrect segmentation, or a result of being unsaid by the speaker (*deletion error*); or phonological context may have caused the segment to be identified as a different phoneme type (*substitution error*).

The basic matching techniques involves aligning the phonemic spelling of the word to be matched with the segmental (phone-like) labels while allowing for the possibility that some of the above types of error may have occurred. Alignment is usually based on the notion of "anchor points" in which stressed vowels and sibilants which are much less likely to be missed are aligned first, followed by the alignment of the remaining vowels and consonants. Once the alignment is completed, the degree of similarity between the word and the unknown phonemic string is defined as a weighted sum of the individual phoneme versus segment label similarity values. These similarity measures are usually available as a confusion matrix generated from a set of manually labeled training data.

Stochastic Matching: This type of matching is used in the IBM system [72], [150], [172] and in the Dragon system [7]. Given a finite-state representation of alternative pronunciations of a word with associated transition probabilities, a dynamic programming technique is used to perform matching left-to-right in a best-first manner. The best phonemic match and the corresponding likelihood are determined by matching all the possible phonemic variations of the word with the unknown segmental phoneme string. This technique, being more mathematically tractable, will probably become the standard technique in word matching. However, careful training procedures are needed to establish the transition probabilities in the graph representation of alternative pronunciations.

Analysis-by-Synthesis: Klatt [75] proposes the use of analysis-by-synthesis as the principal technique for word verification. He feels that phonological phenomena such as vowel reduction, flapping, palatalization, etc., are basically generative in nature and cannot be easily captured in terms of analytic rules. Klatt and Stevens' study of spectrogram reading [76] demonstrates the difficulty. In that study only 77 percent of the segments were correctly (or partially correctly) labeled during the analysis phase, while 97 percent of the words were correctly identified during verification.

Some form of analysis-by-synthesis procedure seems essential to transform an abstract representation of a word into an acoustic representation suitable for matching with the acoustic parametrization of the unknown utterance. Klatt gives the structure and description of an analysis-by-synthesis system which can be used as a word verification component. (see also Section IX of Jelinek [72].)

Future Directions: Whether matching must occur at the signal level, as in the case of analysis-by-synthesis, to achieve accurate recognition will depend on the success achieved by the heuristic and stochastic matching techniques. One would also have to investigate the store versus compute tradeoff in determining whether one can store a set of synthesized (or learned) reference patterns as in the case of word recognition systems (with the necessary juncture rules of course) or whether it is necessary to synthesize them each time. This would depend on the degree of variability and the number of reference patterns needed to cover the range of variability expected for each word. A mixed strategy in which those words that exhibit significant variability (such as function words) are verified by analysis-by-synthesis technique while all others are verified by one of the other techniques might be desirable.

V. Task-Dependent Knowledge

Specifying the task to be performed by a speech recognition system involves defining the *vocabulary* to be used, the grammatical structure of legal or acceptable sentences (*syntax*), the meaning and interrelationships of words within a sentence (*semantics*), and the representation and use of context depending on the conversation (*pragmatics*). Some recognition systems choose to ignore one or more of these KS's, partly because the notions of semantics and pragmatics are somewhat ill-defined and ill-understood at present, and partly because not all the KS's are necessary if the task is sufficiently restricted. The tutorial-review paper by Woods [167] presents a clear discussion of the role of syntax and semantics in speech. In this section, we will illustrate how these task-dependent KS's help to restrict and reduce the combinatorial explosion resulting from the error and uncertainty of the choices at the lower levels.

A major component of an SUS is to understand and respond to a message. There is a large body of literature on language understanding that is relevant. The books edited by Minsky [102], Simon [145], Rustin [133], Colby and Shank [138],

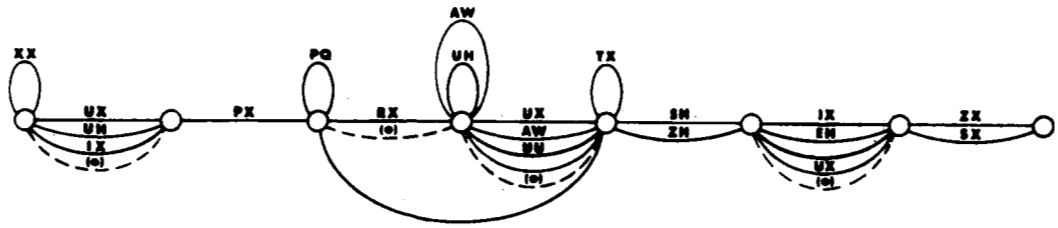

Fig. 24. Phonemic subgraph of the word "approaches" showing possible speaker variation. Branches corresponding to machine error phonemena and indications of second-order constraint have been omitted for clarity of representation. (From Paul *et al.* [114].)

Task	Size of Vocabulary	Language Entropy	Language EQV. Branching factor	Confusability Entropy	Confusability EQV. Branching factor
Digits	10	3.32	10	0.24	1.18
Alphabet	26	4.70	26	2.43	5.39
Alpha-digit	36	5.17	36	2.29	4.89
Chess	31	2.87	7.30	1.37	3.32
Lincoln	237	2.84	7.18		
Extended	411	3.36	12.61		
IBM	250	2.872	7.32		
Prog. language (no syntax)	37	5.21	37.00	1.92	3.78

Fig. 25. Confusability of some common vocabularies. (From Goodman [54].) The first two columns show the entropy per word (i.e., average number of bits it takes to represent a word in the language) and the equivalent branching factor. The last two columns show the confusability of the same vocabulary in terms of entropy (of a noisy channel) and branching factor. Note that the confusability of the digits vocabulary is low while the confusability of the alphabet ("aye," "bee," . . .) is much higher. Note that the average branching factors of the languages used by the systems shown in Figs. 9 and 11 are all less than 10 except for the extended Lincoln task.

Shank [139], and Bobrow and Collins [19] contain a number of the important papers in the natural language understanding area. In this section, we will restrict ourselves to those aspects of the task-dependent knowledge that are directly relevant to the problem of recognition of the utterance.

A. Vocabulary

The primary source of restriction in most speech recognition systems is the vocabulary. Performance of a system is not only affected by the size and dialectal variations of the vocabulary but also by the confusability among the words. The main design choice associated with this level is the representation of alternative pronunciations of the words. If the task permits, one might also wish to select words so as to minimize the confusability among them.

Representation of Phonemic Variation: Most of us know that words are pronounced differently in different contexts. The phrases "David and Robert" and "Fish'n chips" illustrate two different pronunciations of "and." The word "mostly" is sometimes pronounced without the /t/, as "mosly." What is not commonly realized is that much of this variability is rule governed and can be predicted by a set of phonological rules [24], [112]. Starting with a phonemic base form, one can create all possible alternative pronunciations. Early attempts represented each phonemic variation as a separate entry in the lexicon, but most present systems use a more compact network representation. Fig. 24 [114] illustrates the representation of many alternate pronunciations of "approaches" in the network form. Stress and syllable boundary information is also usually entered in the lexicon.

The lexical entries are sometimes preanalyzed to determine all the words that have the same syllable type and represented as an inverted dictionary where one can look up all the words that have the same syllable. This type of representation is useful in generating word hypotheses based on the phones and syllables observed in the symbolic representation of the unknown utterance.

Ambiguity: We saw in Section II how a system which gives 99 percent accuracy on a 200-word multisyllable vocabulary can drop to 89 percent accuracy on a 36-word alpha-digit list. This is because the letters of the alphabet (when pronounced as "aye," "bee," "cee," · · ·) are highly confusable. It is thus important to know not only the size of a vocabulary but also a measure of its confusability. Goodman [54] has studied the confusability of several vocabularies using both theoretical and experimental methods. Fig. 25 summarizes some of his results for several task domains. The first two columns show the entropy per word (i.e., average number of bits required to represent a word in the language) and equivalent branching factor. Thus for the digit sequence recognition task, where any of the 10 digits can follow at every choice point, the average branching factor is 10. However, the confusability of the digits vocabulary is not very high. Using the notion of entropy of a noisy channel, Goodman shows that, from a confusability point of view, the average branching factor for the digit task is only 1.18. For the spelling task (i.e., alphabet recognition task), the confusability is much higher, with a branching factor of 5.39.

Effect of Vocabulary Size on Accuracy: There have been no systematic studies in this area. The Lincoln system performance drops from 49 percent sentence accuracy to 28 percent when the vocabulary increases from 237 words to 411 words. (Note that the complexity of the language, as measured by the average branching factor, has also increased. Fig. 25 shows that the branching factor increased from 7.32 to 12.61—almost proportional to the increase in vocabulary in this case.) The Hearsay-I system shows a similar drop in performance in going from 30 to 194 words. What seems to be important is not so much the size but rather the confusability. In vocabularies that have not been carefully preselected, i.e., they might be assumed to have about the same percentage of words that are confusable, doubling the size of vocabulary seems to double the error rate. This linear increase is contrary to the earlier expectations that, as the vocabularies get larger, confusability among words (and hence the error rate) would only grow less than linearly. However, this linear increase should not be of concern if CSR systems can approach the accuracies being achieved by word recognition systems, i.e., greater than 99 percent accuracy at the word level.

Future Directions: Assuming that current trends in research and technology will continue, there is no reason, in principle at least, why we should not look toward *unlimited vocabulary recognition* systems. However, this will require significant advances in dictionary representation, word hypothesis, and word verification. There are several large phonemic dictionaries of English available in computer readable form [57]. Words that do not exist can be added using grapheme-to-phoneme translation techniques (see the paper by Jon Allen in this issue). Special techniques have to be devised to produce compact and easily retrievable representations of a dictionary containing on the order of a million entries.

Unstressed function words will always be a problem, whether we have a 1000-word system or an unlimited vocabulary system. To establish the feasibility of unlimited vocabulary recognition, we need to consider only the problems of locating and recognizing the rest of the words which can be expected to have at least one stressed syllable. Given the present and projected performances of word hypothesis procedures, fewer than one thousand of the million words are likely to be hypothesized around each stressed syllable. This candidate list can be further pruned to 10 or 20 words using stochastic word verification procedures discussed in Section IV. At this point the remaining words will have fine differences, such as between the words *sit* and *slit*. These will require analysis-by-synthesis, matching with prestored reference patterns for syllables, or some such technique. Any further ambiguities at this stage and prediction and detection of unstressed function words will require the active mediation of higher level processes, such as pragmatics.

B. Syntax

The grammatical structure of sentences can be viewed as principally a mechanism for reducing search by restricting the number of acceptable alternatives. Given a vocabulary of size N, if one permits any word to follow any other word such as "sleep roses dangerously young colorless," the number of possible sentences would be of the order N^L for utterances of less than L words in length. Syntactic structure imposes an ordering and mutually interdependent relationships among words such that only a subset of the N^L is in fact possible.

For example, the IBM New Raleigh task [10] containing 250 words permits only 1.4×10^7 sentences of the possible 250^8 ($\sim 10^{19}$), thereby reducing the search space by a factor of 10^{12}. A more meaningful measure for CSR systems is the average branching factor of the grammar, i.e., the average number of alternative word hypotheses possible at each point in the grammar. For the IBM New Raleigh task, this is about 8 out of the possible 250 words (with a maximum branching factor of about 24). Baker and Bahl [10] measure the complexity of a grammar in terms of the entropy of a word in the language. For the IBM task, the entropy was 2.87. (Note 2^{entropy} is also a measure of the average branching factor of the grammar.) One must also consider the confusability among the alternative words at each choice point. Fig. 25 (from [54]) shows the entropy and the average branching factor that can be expected as a result of the confusability of the vocabulary. Note that the average branching factors for several of the systems discussed in Section II are all less than 10.

Woods [165] proposes four categories to measure the constraint provided by a given grammar. The first two categories are finite-state grammars, the first one having a small branching factor (usually less than 30 at each choice point) and the second one having large branching factors (greater than 400 words at some choice points). Category III systems are artificial languages characterized by context-free grammars permitting recursion, and having large branching factors. Category IV systems are approximations to natural English grammars with considerably larger search space than even category III. Most systems built to data tend to be of the category I type.

Robinson [129] argues that the representation of grammar based on written language is likely to be of limited use. She proposes that the discovery of rules governing spoken language behavior and the development of performance grammars should be based on systematic study of conversational speech. She demonstrates some of the strategies and constraints operating in performance (and the rule-governed regularities they produce) by analysis of several task-oriented dialogs.

Representation: The most commonly used representation for grammars is some form of network representation. Woods [167] uses augmented transition network (ATN) grammars. An ATN consists of a finite-state transition diagram with embedded recursion added, and a set of registers which can hold arbitrary pieces of tree structure and can be modified depending on the actions (programs) associated with the arcs of the grammar. The ATN formalism is known to have the linguistic adequacy of a transformational grammar while providing more of the efficiency needed in parsing algorithms. The papers by Bates [15], [16], and Paxton [115] contain examples of the representation and use of transition network grammars.

Baker [7] and Jelinek, Bahl, and Mercer [71] view grammar as a probabilistic function of a Markov process and represents it as a finite-state network with transition probabilities including the self-transition probability (providing for recursion). Fig. 26 gives an illustration of Baker's representation of the chess grammar (transition probabilities not shown). This representation permits the Dragon system to determine the optimal match for the utterance using the dynamic programming algorithm.

C. Semantics

The term *semantics* means different things to different people. Here we use the term to denote the rules and relationships associated with the meaning of symbols. By this we mean the rules of language which tell us that the sequence of words "colorless yellow ideas" is not meaningful. It is not always possible to detect semantic inconsistency by looking at adjacent word pairs as in the previous example. The statements "Give me about a yard" and "Tell me about Tom" are acceptable, but "Give me about Tom" is not. Given a set of possibly disjoint words, semantic information must determine whether they are compatible and meaningful. The role of semantics in the recognition aspects of an SUS is analogous to that of syntax, i.e., it provides a mechanism for reducing search by restricting the number of acceptable alternatives. The papers by Woods [167], Nash-Webber [103], and Hendrix [60] illustrate the uses of semantic knowledge in speech understanding systems.

The principal technique used to represent this KS is a *semantic net*. Fig. 27 gives the structure of a semantic net fragment used by Hendrix [61]. A semantic net is used to represent objects, relationships among objects, events, rules, and situations. Such a net can be used to predict or verify possible word hypotheses in an unknown utterance. Given the concepts of *content* and *contains* and the meaning of the word "in," one may accept the statement "The bolts are in the box" and reject "The box is in the bolts" using such a semantic net.

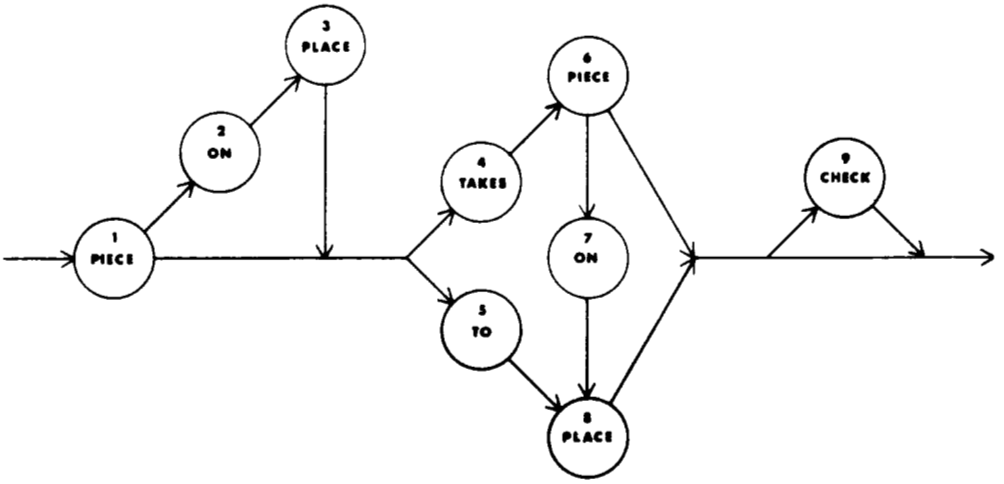

Fig. 26. Fragment of network grammar for Chess. (From Baker [7].)

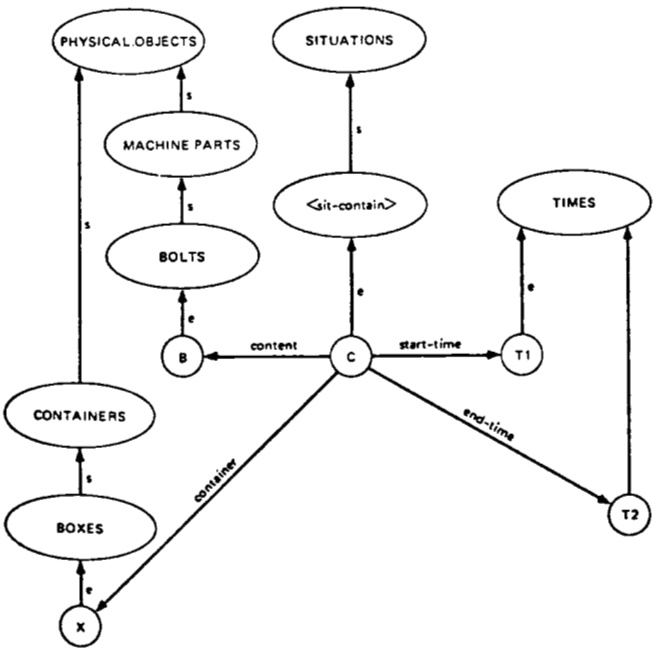

Fig. 27. Part of a typical semantic net. (From Hendrix [61].)

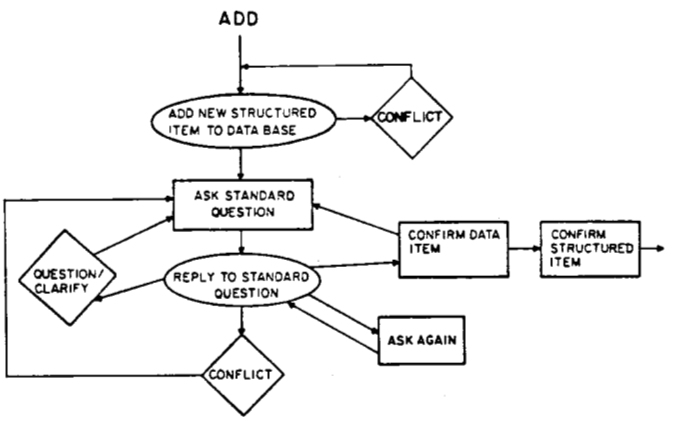

Fig. 28. User-oriented discourse model. (From Woods *et al.* [169].)

The chess task used by Hearsay-I [123] illustrates how knowledge about the partially recognized utterance can be used to constrain the hypotheses to a small set of words. For example, given that a word such as "captures" or "takes" appears in the partially recognized utterance, this information can be used to further restrict the search to the capture moves in a particular board position. This restricted set of moves is used to give high semantic preference to the key content words that may occur in the capture moves.

D. Pragmatics and Discourse Analysis

The term *pragmatics* usually leads to even more confusion and misunderstanding than *semantics*. Here we use it to mean conversation-dependent contextual information, i.e., task-related information accumulated so far through man–machine dialog. At a given point in the conversation, the user may use an elided (or non-well-formed) sentence or may use pronominal reference to a previous subject. For example, consider the sequence of questions, "How much does Tom weigh? How about John? What is his height?" It is obvious that in the second question one is asking for John's weight even though it does not appear explicitly. In the third question, does the term "his" refer to Tom or John? It is ambiguous, but the most plausible interpretation is that it refers to John, the subject of the immediately preceding utterance. Interpretation and validation of such questions based on pragmatics of the situation require the representation of dialog so that missing constituents can be inferred. Deutsch [29] uses a tree structure representation scheme for handling simple forms of ellipsis and anaphora.

The other main role of pragmatics is to predict the user behavior (user model) based on the dialog. Fig. 28 is a transition network diagram indicating the common modes of interaction found in travel budget management dialog [167]. It illustrates various possible modes of interaction of the user such as: user adds a new item to the data base, system points out contradiction, user asks a question, system answers, user makes a change, and so on. If the pragmatic knowledge can accurately ascertain the state of the user, it can be used to achieve syntactic subselection, i.e., only a small set of all the possible grammatical constructs would be likely in that situation. We do not yet have any system which has made effective use of user models, but several are attempting to do so.

VI. System Organization

System organization is a catch-all term that describes the art of transforming ideas into working programs. Given that many of the problems of connected speech recognition and

understanding are not well defined, the issue of system organization assumes added importance. We know that all the available sources of knowledge must communicate and cooperate in the presence of error and uncertainty. We do not know how to do it effectively or efficiently. The system must work smoothly with high-data-rate real-time input and provide facilities for speaker- and task-dependent knowledge acquisition. The tutorial-review paper by Reddy and Erman [122] and the papers by Newell [107], Baker [9], Barnett [14], Erman [35], Fennell [43], Jelinek *et al.* [71], Jelinek [72], and Lesser [85] provide more detailed discussions of system organization issues. In this section, we will briefly review the problems of system organization.

Many of the principal issues of system organization were raised and discussed in Section II while studying the structure of various types of speech recognition systems and examining what makes such systems difficult to realize. In particular, we have seen how various systems were organized to permit several diverse sources of knowledge to communicate and cooperate, how search strategies were devised to deal with error and ambiguity, and how knowledge is represented and used. In this section, we will discuss some of the related issues of system organization which were not covered earlier.

A. Control Strategies in the Presence of Ambiguity and Error

There are several sources of error and ambiguity in speech recognition. In spontaneous (nonmaximally differentiated) connected speech many expected features (and phones) may be missing. Variability due to noise and speaker leads to errors. Incomplete and/or inaccurate KS's at each level introduce more errors. In simple hierarchical systems, these errors propagate through various levels, compounding the error rate. Thus every system organization must cater to the inevitability of errors and handle them in a graceful manner.

Given the errorful nature of speech processing, one has to consider several alternative hypotheses (or interpretations) since the hypothesis with the highest rating may not be the correct one. In Fig. 12, we see how the problem of error is transformed into a problem of uncertainty by considering several plausible alternative hypotheses. At that point the problem becomes one of search through this multilevel network to discover the best path that is consistent with all the KS's. There are several search techniques developed in the field of artificial intelligence that become potentially useful. We will consider two of these techniques that have been used in speech recognition.

The commonly used strategy is *best-first search*. This technique is used in Hearsay-I, Lincoln, and in a modified form in the IBM system (see Section II-B and Section V of Jelinek [72]). This technique is best explained by an example [122]. In Fig. 29, we see a tree of possible alternatives that had arisen in the analysis of the utterance "Are your headaches severe?" We find that there are nine alternatives for the first word: "have," "are," "where," and six others. The ratings indicating the likelihood, which can be derived either mathematically [71] or heuristically [55], are given under each word. Given that the word "have" has the highest rating (470), we begin to explore that path. "Have" is followed by a single alternative "you." The combined rating for the sequence "have you" is given under "you." The rating of 455 makes the sequence "have you" better than the other alternative paths. Proceeding along this path, we have three alternative words that can follow "have you." The sequence "have you had" receives the highest rating of 447 but is no longer the highest rated path. Search is suspended along this path and we begin exploring the alternatives that can follow "are." Of the three alternatives, the path "are your" has the highest rating of 453. Proceeding along this path, we get a rating of 445 for the sequence "are your headaches." Since this is lower than 447, we now suspend this search and resume search along the path "have you had." The highest rated sequence "have you had mumps" is lower than 445, so we resume the path of "are your headaches" again. Now we find that "are your headaches severe" has the highest rating of 462 and, being the end of a sentence, cannot proceed any further. We accept this as the most probable sentence and return it as the answer to the search.

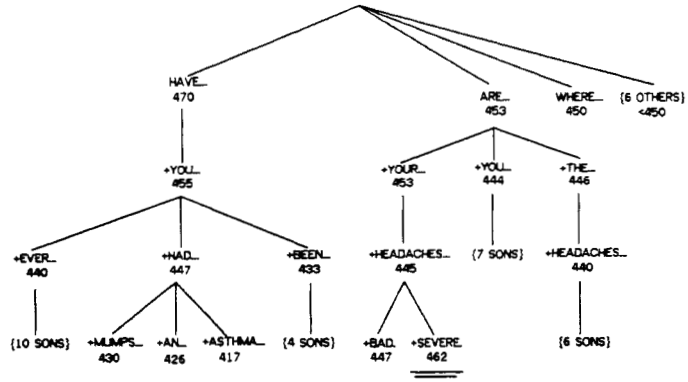

Fig. 29. Example of best-first search. (From Reddy and Erman [122].)

Another technique that has been used in speech recognition is to *search all possible paths in parallel* but constrain the search only through those sequences which are valid paths in a specific network [7], [8]. This network is constructed to be an integrated representation of all the available KS's. Exponential growth is constrained through the use of a Markov assumption which limits the relevant context and collapses many alternative sequences to a single state. With this technique, search time is linear in the number of states in the network and in the length of the utterance.

There are several other search techniques, such as prosodically guided search [81], anchor points, focus of attention [86], a few alternative paths in parallel [90], and so on. The relative advantages and disadvantages of these techniques are not clearly understood.

B. Real-Time Input

Unlike most other forms of computer input, speech is critically *data directed*. That is, initiation and termination of the input depend on the incoming data rather than on program control. Thus a system must be prepared to continuously monitor the speech input device (analog-to-digital converter, filter bank, or what have you) to determine if the signal is speech or noise.

The *high data rates* associated with speech input (100 to 300 kbit/s) imply that a system cannot afford to have resident in primary memory more than a few seconds of speech data (usually no more than a single utterance). Thus the data must be immediately processed, placed in secondary storage, or played back. Keeping two high-data-rate devices serviced is not a major problem if the system is dedicated. However, if the system has a general-purpose operating system, special care must be taken to see that the device service overhead is

low in order to avoid loss of data. This often becomes difficult to achieve with two active devices because of the costs associated with process synchronization, buffer service routines, and signal detection [35]. System primitives available within the system for performing these operations tend to be too slow and need to be reprogrammed.

C. Knowledge Acquisition

Every speech recognition system uses thresholds, templates, probabilities, and so on, based on measurements obtained on training data. In the case of word recognition systems, the problem is solved neatly since reference patterns capture noise, microphone, speaker, and phonological variability in a single step. In CSR systems, however, one has to deal with most of these sources of variability individually.

The first question that arises is what parameters will be seen when a sound is spoken. This depends not only on the parametric representation used but also on the speaker, the microphone used, and the noise in the environment. Systems using formant representation attempt to normalize speaker dependencies by estimating the shifts in formant trajectories, the length of the vocal tract of the speaker, etc. Systems that use prototype template matching require spectral templates (or some other parametric representation) for each sound to be recognized for each speaker. These in turn require a set of carefully manually segmented and labeled sentences for each speaker. Machine-aided segmentation and labeling has been attempted [6], but this requires a reasonable set of starting templates (the chicken and the egg problem).

The second question that arises is what phonemes are observed when a word (or a sequence of words) is spoken. Phonological rules predict some phenomena but they do not predict (at least not as yet) that a stop in some context can be three-quarters voiced and one-quarter unvoiced. Some systems attempt to accumulate such acoustic-phonetic phenomena from real data. Again, one needs manually labeled data or machine-labeled data for training of the system. Baker and Bahl [10] present the results of an interesting training method for automatically learning transition probabilities described in [71] and [72].

Many of the heuristic techniques used by various systems require substantial amounts of labeled data. This in turn necessitates the design of several interactive programs with graphical output for collection and validation of rules. Labeled data are also essential for systematic performance analysis of segmentation, labeling, syllable detection, word hypothesis, and word verification procedures.

VII. Conclusions

We have attempted to review the recent developments in speech recognition. The focus has been to review research progress, to indicate the areas of difficulty, why they are difficult, and how they are being solved. The paper is not intended as a survey of all known results in speech recognition and represents only one point of view of important issues, problems, and solutions.

The past few years have seen several conceptual and scientific advances in the field. We have already discussed many of these aspects earlier in the text. We will summarize them here.

1) For the first time we have available extensive analysis of connected speech. We know connected speech recognition is not impossible.
2) The role and use of knowledge are better understood. Almost all systems use knowledge to generate hypotheses and/or verify them.
3) Error and ambiguity can be handled within the framework of search.
4) Stochastic representations and dynamic programming provide a simple and effective solution to the matching problem.
5) Network representation of knowledge is a compact and computationally efficient way of generating and verifying hypotheses.
6) For the first time we have some techniques for the codification and use of phonological rules in speech recognition systems.
7) For the first time we have effective tools for the study of prosodics and the application of prosodic information to speech recognition.
8) There has been comparatively significant progress in the past five years in the areas of parameter extraction, formant tracking, feature detection, segmentation, and labeling.
9) Linear predictive coding and Itakura's distance metric represent effective digital techniques for analysis and matching at the signal level.

In spite of the significant progress, there are still several unsolved problem areas: signal processing associated with noise, telephone and speaker normalization, real-time live input providing graceful interaction with the user, careful and systematic performance analysis of the existing systems, and labeled data bases. In addition, continued progress is necessary in almost all the knowledge domains to establish optimal representation and use of different sources of knowledge.

We have indicated that it may be possible to build a $1000 isolated word recognition system, a $20 000 connected speech system, an unlimited vocabulary system, and so on. All of these seem feasible and can probably be realized within the next 10 years if the present momentum in speech recognition research can be continued. However, they may never be realized without significant and directed research effort. We are still far from being able to handle relatively unrestricted dialogs from a large population of speakers in uncontrolled environments. Many more years of intensive research seems necessary to achieve such a goal.

Acknowledgment

The author wishes to thank his colleagues L. Erman and A. Newell for their many valuable comments on various drafts of this paper, and also J. K. Baker, J. M. Baker, R. Dixon, J. Forgie, H. Goldberg, J. Shoup-Hummel, W. Lea, F. Jelinek, T. Martin, N. Neuburg, L. Rabiner, A. Rosenberg, H. Silverman, G. White, and J. Wolf for their comments on various parts of this paper.

References

[1] B. S. Atal and M. R. Schroeder, "Predictive coding of speech signals," in *Proc. 1967 Conf. Speech Communication and Processing*, pp. 360–361, Nov. 1967.
[2] ——, "Predictive coding of speech signals," in *Proc. Int. Congr. Acoustics*, C-5-4, Tokyo, Japan, Aug. 1968.
[3] ——, "Adaptive predicting coding of speech signals," *Bell Syst. Tech. J.*, vol. 49, no. 6, pp. 1973–1986, 1970.
[4] B. S. Atal and S. L. Hanauer, "Speech analysis and synthesis by linear prediction of the speech wave," *J. Acoust. Soc. Amer.*, vol. 50, no. 2, pp. 637–655, 1971.
[5] B. S. Atal, "Linear prediction of speech—Recent advances with applications to speech analysis," in [121, pp. 221–230].

[6] J. K. Baker, "Machine-aided labeling of connected speech," in *Working Papers in Speech Recognition II*, Tech. Rep., Comput. Sci. Dep., Carnegie-Mellon Univ., Pittsburgh, PA, 1973.
[7] —, "The DRAGON system—An overview," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 24–29, Feb. 1975.
[8] —, "Stochastic modeling as a means of automatic speech recognition," Ph.D. dissertation, Computer Sci. Dep., Carnegie-Mellon Univ., Pittsburgh, PA, 1975.
[9] —, "Stochastic modeling for automatic speech understanding," in [121, pp. 500–520].
[10] J. K. Baker and L. Bahl, "Some experiments in automatic recognition of continuous speech," in *Proc. 11th Annu. IEEE Computer Society Conf.*, pp. 326–329, 1975.
[11] J. M. Baker, "A new time-domain analysis of human speech and other complex waveforms," Ph.D. dissertations, Carnegie-Mellon Univ., Pittsburgh, PA, 1975.
[12] R. Bakis, "Continuous speech recognition via centisecond acoustic states," *J. Acoustic Society Amer.*, vol. 58 (to be presented at the 91st meeting of ASA).
[13] J. A. Barnett, "A vocal data management system," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 185–188, June 1973.
[14] —, "Module linkage and communication in large systems," in [121, pp. 500–520].
[15] M. Bates, "The use of syntax in a speech understanding system," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 112–117, Feb. 1975.
[16] —, "Syntactic analysis in a speech understanding system," Ph.D. dissertation, Harvard Univ., Cambridge, MA, 1975. Also Rep. 3116, Bolt Beranek and Newman, Inc., Cambridge, MA, 1975.
[17] B. Beek, E. P. Neuburg, D. C. Hodge, R. S. Vonusa, and R. A. Curtis, "An assessment of the technology of automatic speech recognition for military application," NATO Rep. To be published by Rome Air Development Center, Rome, NY, 1976.
[18] D. G. Bobrow and D. H. Klatt, "A limited speech recognition system," in *Proc. AFIPS Fall Joint Computer Conf.*, vol. 33. Washington, DC: Thompson, 1968, pp. 305–318.
[19] D. G. Bobrow and A. Collins, *Representation and Understanding*. New York: Academic Press, 1975.
[20] R. Breaux and I. Goldstein, "Development of machine speech understanding for automated instructional systems," in *Proc. 8th NTECL/Industry Conf.*, Naval Training Equipment Center, Orlando, FL, Nov. 1975.
[21] D. J. Broad and J. E. Shoup, "Concepts for acoustic phonetic recognition," in [121, pp. 243–274].
[22] E. C. Carterette and M. H. Jones, *Informal Speech*. Berkeley, CA: Univ. of California Press, p. 16 ff, 1974.
[23] A. Chapanis, "Interactive human communication," *Sci. Amer.*, vol. 232, no. 3, pp. 36–42, 1975.
[24] P. S. Cohen and R. L. Mercer, "The phonological component of an automatic speech recognition system," in [121, pp. 275–320].
[25] E. E. David, Jr., and P. B. Denes, *Human Communication: A Unified View*. New York: McGraw-Hill, 1972.
[26] R. De Mori, S. Rivoira, and A. Serra, "A speech understanding system with learning capability," in *Proc. 4th Int. Joint Conf. Artificial Intelligence*, Tbilisi, USSR, 1975.
[27] M. Derkach, "Heuristic models for automatic recognition of spoken words," Quart. Progr. Status Rep., Speech Transmission Labs., KTH, Stockholm, Sweden, pp. 39–49, Jan–Mar. 1970.
[28] M. Derkach, R. Gumetsky, B. Gura, and L. Mishin, "Automatic recognition of simplified sentences constructed of the limited lexicon," in [41].
[29] B. G. Deutsch, "Discourse analysis and pragmatics," in [156, ch. VI].
[30] N. R. Dixon and H. F. Silverman, "A description of a parametrically controlled modular structure for speech processing," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 87–91, Feb. 1975.
[31] N. R. Dixon and H. F. Silverman, "A general language-operated decision implementation system (GLODIS): Its application to continuous speech segmentation," (to appear in *IEEE Trans. Acoust. Speech, Signal Processing*, vol. ASSP-24, 1976.
[32] —, "Some encouraging results for general purpose continuous speech recognition," in *Proc. 1975 Int. Conf. Cybernetics and Society*, San Francisco, CA, pp. 293–295, 1975.
[33] P. H. Dorn, "Whither data entry," *Datamation*, vol. 27, pp. 49–51, Mar. 1973.
[34] L. D. Erman, Ed., *Contributed Papers of IEEE Symp. Speech Recognition*, Carnegie-Mellon Univ., Pittsburgh, PA (IEEE Cat. 74CHO878-9 AE), 1974.
[35] —, "An environment and system for machine understanding of connected speech," Ph.D. dissertation, Comput. Sci. Dep., Stanford Univ., Tech. Rep., Comput. Sci. Dep., Carnegie-Mellon Univ., Pittsburgh, PA, 1974.
[36] —, "Overview of the Hearsay speech understanding research," *Comput. Sci. Res. Rev.*, Comput. Sci. Dep., Carnegie-Mellon Univ., Pittsburgh, PA, 1975.
[37] L. D. Erman and V. R. Lesser, "A multi-level organization for problem solving using many, diverse cooperating sources of knowledge," in *Proc. 4th Int. Joint Conf. Artificial Intelligence*, Tbilisi, USSR, 1975.
[38] L. D. Erman, and D. R. Reddy, "Implications of telephone input for automatic speech recognition," in *Proc. 7th Int. Congr. Acoustics*, vol. 3, Budapest, Hungary, 1971, pp. 85–88.
[39] C. G. M. Fant, "Automatic recognition and speech research," Quart. Progr. Status Rep., Speech Transmission Labs., KTH, Stockholm, Sweden, pp. 16–31, Jan–Mar. 1970.
[40] —, *Speech Sounds and Features*. Cambridge, MA: M.I.T. Press, 1973.
[41] —, Ed., *Proc. Speech Communications Seminar*, Stockholm, Sweden, 1974. To be published by Almquist and Wiksell Int. (Stockholm) and Wiley (New York), 1975.
[42] C. G. M. Fant and B. Lindblom, "Studies of minimal speech sound units," Quart. Progr. Status Rep., Speech Transmission Labs., KTH, Stockholm, Sweden, pp. 1–11, Apr.–June 1961.
[43] R. D. Fennell, "Multiprocess software architecture for AI problem solving," Ph.D. dissertation, Comput. Sci. Dep., Carnegie-Mellon Univ., Pittsburgh, PA, 1975.
[44] J. L. Flanagan, *Speech Analysis, Synthesis and Perception*. New York: Springer, 1972.
[45] J. W. Forgie, personal communication, 1975.
[46] J. W. Forgie and C. D. Forgie, "Results obtained from a vowel recognition computer program," *J. Acoust. Soc. Amer.*, vol. 31, pp. 1480–1489, 1959.
[47] J. W. Forgie *et al.*, "Speech understanding systems—semiannual Tech. Summary Rep.–May 1974," M.I.T. Lincoln Lab., Lexington, MA, 1974.
[48] J. W. Forgie, D. E. Hall, and R. W. Wiesen, "An overview of the Lincoln Laboratory speech recognition system," *J. Acoust. Soc. Amer.*, vol. 56, S27 (A), 1974.
[49] L. J. Gerstman, "Classification of self-normalized vowels, *IEEE Trans. Audio Electroacoust.*, vol. AU-16, pp. 78–80, Mar. 1968.
[50] R. A. Gillmann, "A fast frequency domain pitch algorithm," System Development Corp., Santa Monica, CA, unpublished rep., 1975.
[51] B. Gold, "Word recognition computer program," M.I.T. Lincoln Lab., Cambridge, MA, Tech. Rep. 456, 1966.
[52] B. Gold and L. Rabiner, "Parallel processing techniques for estimating pitch periods of speech in time domain," *J. Acoust. Soc. Amer.*, vol. 46, no. 2, pp. 422–449, 1969.
[53] H. G. Goldberg, "Segmentation and labeling of speech: A comparative performance evaluation," Ph.D. dissertation, Computer Science Department, Carnegie-Mellon University, Pittsburgh, PA, 1975.
[54] R. G. Goodman, "Language design for man–machine communication," Ph.D. dissertation (in preperation), Comput. Sci. Dep., Stanford Univ., Stanford, CA, 1976.
[55] D. E. Hall, "The linguistic module, Final report on speech understanding systems," M.I.T. Lincoln Lab., Lexington, MA, unpublished rep., 1974.
[56] D. E. Hall and J. W. Forgie, "Parsing and word matching in the Lincoln Laboratory speech recognition system," *J. Acoust. Soc. Amer.*, vol. 56, S27 (A), 1974.
[57] E. Hayden and J. Shoup, "SCRL computerized pronouncing dictionary," Speech Communication Res. Lab., Santa Barbara, CA, unpublished, 1975.
[58] F. Hayes-Roth and D. J. Mostow, "An automatically compilable recognition network for structured patterns," in *Proc. 4th Int. Joint Conf. Artificial Intelligence*, Tbilsi, USSR, 1975.
[59] F. Hayes-Roth and V. Lesser, "Focus of attention in a distributed logic speech understanding system," Comput. Sci. Dep., Carnegie-Mellon Univ., Pittsburgh, PA, Tech. Rep., 1976.
[60] G. G. Hendrix, "Expanding the utility of semantic networks through partitioning," in *Proc. 4th Int. Joint Conf. Artificial Intelligence*, Tbilisi, USSR, 1975.
[61] —, "Semantics," in [156, ch. V].
[62] C. Hewitt, "Description and theoretical analysis, (using schemata) of Planner: A language for proving theorems and manipulation models in a robot," M.I.T. Project MAC, Cambridge, MA, AI Memo. 251, 1972.
[63] D. R. Hill, "Man–machine interaction using speech," in *Advances in Computers*, F. L. Alt, M. Rubinoff, and M. C. Yovits, Eds., vol. II. New York: Academic Press, 1971, pp. 165–230.
[64] A. D. C. Holden, E. Strasbourger, and L. Price, "A computer programming system using continuous speech input," in [34].
[65] J. F. Hemdal and G. W. Hughes, "A feature based computer recognition program for the modeling of vowel perception," in [158, pp. 440–453].
[66] S. R. Hyde, "Automatic speech recognition: Literature, survey, and discussion," in [25].
[67] S. Itahashi, S. Makino, and K. Kido, "Discrete-word recognition utilizing a word dictionary and phonological rules," *IEEE Trans.*

Audio Electroacoust., vol. AU-21, pp. 239–249, June 1973.
[68] F. Itakura and S. Saito, "Analysis synthesis telephony based on the maximum likelihood method," in *Proc. 6th Int. Congr. Acoustics*, 1968, Paper C-5-5.
[69] ——, "A statistical method for estimation of speech spectral density and formant frequencies," *Electron. Commun. Japan*, vol. 53-A, no. 1, pp. 36–43, 1970.
[70] F. Itakura, "Minimum prediction residual principle applied to speech recognition," *IEEE Trans. Acoust., Speech, Signal Processing*,vol. ASSP-23, pp. 67–72, Feb. 1975.
[71] F. Jelinek, L. R. Bahl, and R. L. Mercer, "Design of a linguistic statistical decoder for the recognition of continuous speech," *IEEE Trans. Inform. Theory*, vol. IT-21, pp. 250–256, May 1975.
[72] F. Jelinek, "Continuous speech recognition by statistical methods," this issue, pp. 532–556.
[73] I. Kameny, "Comparison of the formant spaces of retroflexed and nonretroflexed vowels," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 38–49, Feb. 1975.
[74] D. H. Klatt, "On the design of a speech understanding system," in [41].
[75] ——, "Word verification in a speech understanding system," in [121, pp. 321–341].
[76] D. H. Klatt and K. N. Stevens, "On the automatic recognition of continuous speech: Implications from a spectrogram-reading experiment," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 210–217, June 1973.
[77] J. S. Kriz, "A 16-bit A-D-A conversion system for high-fidelity audio research," *IEEE Trans. Acoust. Speech, Signal Processing*, vol. ASSP-23, pp. 146–149, 1975.
[78] W. A. Lea, "Establishing the value of voice communication with computers," *IEEE Trans. Audio Electroacoust.*, vol. AU-16, pp. 184–197, June 1968.
[79] ——, "Intonational cues to the constituent structure and phonemics of spoken English," Ph.D. dissertation, School of Elec. Eng., Purdue Univ., Lafayette, IN, 1972.
[80] ——, "Prosodic aids to speech recognition: IV—A general strategy for prosodically-guided speech understanding," Sperry Univac Rep. PX10791, 1974.
[81] W. A. Lea, M. F. Medress, and T. E. Skinner, "A prosodically guided speech understanding system," *IEEE Trans. Acoust. Speech, Signal Processing*, vol. ASSP-23, pp. 30–38, Feb. 1975.
[82] W. A. Lea and D. R. Kloker, "Prosodic aids to speech recognition: VI—Timing cues to linguistic structure and improved computer program," Univac Rep. Px11239, 1975.
[83] I. Lehiste, *Readings in Acoustic Phonetics.* Cambridge, MA: M.I.T. Press, 1967.
[84] ——, *Suprasegmentals.* Cambridge, MA: M.I.T. Press, 1970.
[85] V. R. Lesser, "Parallel processing in speech understanding: A survey of design problems," in [121, pp. 481–499].
[86] V. R. Lesser, R. D. Fennell, L. D. Erman, and D. R. Reddy, "Organization of the Hearsay-II speech understanding system," *IEEE Trans. Acoust. Speech, Signal Processing*, vol. ASSP-23, pp. 11–23, 1975.
[87] J. C. R. Licklinder, "Man computer symbiosis," in *Perspectives on the Computer Revolution*, Z. W. Pylyshyn, Ed. Englewood Cliffs, NJ: Prentice-Hall, 1970, pp. 306–318. (First published in *IRE Trans. Hum. Factors Electron.*, vol. HFE-1, 1960).
[88] N. Lindgren, "Machine recognition of human language," *IEEE Spectrum*, vol. 2, Mar., Apr., May, 1965.
[89] ——, "Speech—Man's natural communication," *IEEE Spectrum*, vol. 4, pp. 75–86, June 1967.
[90] B. Lowerre, "A comparative performance analysis of speech understanding systems," Ph.D. dissertation (in preparation), Comput. Sci. Dep., Carnegie-Mellon Univ., Pittsburgh, PA, 1976.
[91] J. Makhoul, "Spectral analysis of speech by linear prediction," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 140–148, June 1973.
[92] ——, "Linear prediction: A tutorial review," *Proc. IEEE (Special Issue on Digital Signal Processing)*, vol. 63, pp. 561–580, Apr. 1975.
[93] ——, "Linear prediction in automatic speech recognition," in [121, pp. 183–220].
[94] J. D. Markel, "Digital inverse filtering—A new tool for formant trajectory estimation," *IEEE Trans. Audio Electroacoust.*, vol. AU-20, pp. 129–137, June 1972.
[95] T. B. Martin, "Acoustic recognition of a limited vocabulary in continuous speech," Ph.D. dissertation, Univ. of Pennsylvania, Philadelphia, 1970.
[96] ——, "Applications of limited vocabulary recognition systems," in [121, pp. 55–71].
[97] S. S. McCandless, "An algorithm for automatic formant extraction using linear prediction spectra," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-22, pp. 135–141, Apr. 1974.
[98] M. Medress, "Computer recognition of single-syllable English words," Ph.D. dissertation, M.I.T., Cambridge, MA, 1969.
[99] P. Mermelstein, "A phonetic-context controlled strategy for segmentation and phonetic labeling of speech," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 79–82, Feb. 1975.
[100] ——, "Automatic segmentation of speech into syllable units," *J. Acoust. Soc. Amer.*, vol. 58, pp. 880–883, 1975.
[101] G. A. Miller and S. Isard, "Some perceptual consequences of linguistic rules," *J. Verbal Learning Behavior*, vol. 2, pp. 217–228, 1963.
[102] M. L. Minsky, Ed., *Semantic Information Processing.* Cambridge, MA: M.I.T. Press, 1970.
[103] B. Nash-Webber, "Semantic support for a speech understanding system," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 124–128, Feb. 1975.
[104] R. B. Neely and D. R. Reddy, "Speech recognition in the presence of noise," in *Proc. 7th Int. Congr. Acoustics*, vol. 3, Budapest, 1971, pp. 177–180.
[105] E. P. Neuburg, "Philosophies of speech recognition," in [121, pp. 83–95].
[106] A. Newell, "Production systems: Models of control structures," in *Visual Information Processing*, W. C. Chase, Ed. New York: Academic Press, 1973, pp. 463–526.
[107] ——, "A tutorial on speech understanding systems," in [121, pp. 3–54].
[108] A. Newell, J. Barnett, J. Forgie, C. Green, D. Klatt, J. C. R. Licklinder, J. Munson, R. Reddy, and W. Woods, *Speech Understanding Systems: Final Report of a Study Group*, 1971. (Reprinted by North-Holland/American Elsevier, Amsterdam, Netherlands, 1973).
[109] A. Newell, F. S. Cooper, J. W. Forgie, C. C. Green, D. H. Klatt, M. F. Medress, E. P. Neuburg, M. H. O'Malley, D. R. Reddy, B. Ritea, J. E. Shoup, D. E. Walker, and W. A. Woods, "Considerations for a follow-on ARPA research program for speech understanding systems," available from Comput. Sci. Dep., Carnegie-Mellon Univ., Pittsburgh, PA, 1975.
[110] R. B. Ochsman and A. Chapanis, "The effects of 10 communication modes on the behavior of teams during co-operative problem-solving," *Int. J. Man-Machine Studies*, vol. 6, pp. 579–619, 1974.
[111] M. M. O'Malley, D. Kloker, and B. Dara-Abrams, "Recovering parentheses from spoken algebraic expressions," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 217–220, June 1973.
[112] B. T. Oshika, V. W. Zue, R. V. Weeks, H. Nue, and J. Aurbach, "The role of phonological rules in speech understanding research," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 104–112, Feb. 1975.
[113] K. W. Otten, "Approaches to the machine recognition of conversational speech," in [63, pp. 127–163].
[114] J. E. Paul, A. S. Rabinowitz, J. P. Riganati, V. A. Vitols, and M. L. Griffith, "Automatic recognition of continuous speech: Further development of a hierarchical strategy," Rome Air Development Center, Rome, NY, RADC-TR-73-319, 1973.
[115] W. H. Paxton, "The parsing system," in [156, ch. III].
[116] J. R. Pierce, "Whither speech recognition?" *J. Acoust. Soc. Amer.*, vol. 46, pp. 1049–1051, 1969.
[117] L. C. Pols, "Real-time recognition of spoken words," *IEEE Trans. Comput.*, vol. C-20, pp. 972–978, Sept. 1971.
[118] L. R. Rabiner and M. R. Sambur, "An algorithm for determining the endpoints of isolated utterances," *Bell Syst. Tech. J.*, vol. 54, no. 2, pp. 297–315, 1975.
[119] D. R. Reddy, "Segmentation of speech sounds," *J. Acoust. Soc. Amer.*, vol. 40, pp. 307–312, 1966.
[120] ——, "Computer recognition of connected speech," *J. Acoust. Soc. Amer.*, vol. 42, pp. 329–347, 1967.
[121] ——, Ed., *Speech Recognition: Invited Papers of the IEEE Symp.* New York: Academic Press, 1975.
[122] D. R. Reddy and L. D. Erman, "Tutorial on system organization for speech understanding," in [121, pp. 457–479].
[123] D. R. Reddy, L. D. Erman, and R. B. Neely, "A model and a system for machine recognition of speech," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 229–238, June 1973.
[124] D. R. Reddy and A. Newell, "Knowledge and its representation in a speech understanding system," in *Knowledge and Cognition*, L. W. Gregg, Ed. Washington, DC: L. Earlbaum Assoc., 1974, pp. 253–285.
[125] D. R. Reddy and P. J. Vicens, "A procedure for segmentation of connected speech," *J. Audio Eng. Soc.*, vol. 16, pp. 404–412, 1968.
[126] C. W. Reedyk, "Noise-cancelling electret microphone for lightweight head telephone sets," *J. Acoust. Soc. Amer.*, vol. 53, pp. 1609–1615, 1973.
[127] B. Ritea, "Automatic speech understanding systems," in *Proc. 11th Annu. IEEE Computer Society Conf.*, Washington, DC, Sept. 1975.
[128] J. J. Robinson, "The language definition," in [156, ch. IV].
[129] ——, "Performance grammars," in [121, pp. 401–427].
[130] P. J. Rovner, J. Makhoul, J. Wolf, and J. Colarusso, "Where the

words are: Lexical retrieval in a speech understanding system," in [34, pp. 160–164].
[131] P. J. Rovner, B. Nash-Webber, and W. Woods, "Control concepts in a speech understanding system," *IEEE Trans. Acoust., Speech, Signal, Processing*, vol. ASSP-23, pp. 136–140, Feb. 1975.
[132] J. F. Rulifson *et al.*, "QA4: A procedural calculus for intuitive reasoning," AI Center, Stanford Res. Inst., Menlo Park, CA, Tech. Note 73, 1973.
[133] R. Rustin, Ed., *Natural Language Processing*. New York: Algorithmics Press, 1973.
[134] T. Sakai and S. Nakagawa, "Continuous speech understanding system LITHAN," Dept. Inform. Sci., Kyoto Univ., Kyoto, Japan, Tech. Rep., 1975.
[135] M. R. Sambur and L. R. Rabiner, "A speaker-independent digit-recognition system," *Bell Syst. Tech. J.*, vol. 54, pp. 81–102, 1975.
[136] R. Schwartz and J. Makhoul, "Where the phonemes are: Dealing with Ambiguity in acoustic-phonetic recognition," *IEEE Trans. Acoust., Speech, Signal, Processing*, vol. ASSP-23, pp. 50–53, Feb. 1975.
[137] R. N. Schafer and L. R. Rabiner, "Parametric representations of speech," in [121, pp. 99–150].
[138] R. C. Shank and K. M. Colby, *Computer Models of Thought and Language*. San Francisco, CA: Freeman, 1973.
[139] R. C. Shank, Ed., *Conceptual Information Processing*. Amsterdam, Netherlands: North-Holland, 1975.
[140] D. E. Shanon, "Prediction and entropy of printed English," *Bell Syst. Tech. J.*, vol. 30, pp. 50–64, 1951.
[141] J. N. Shearme and P. F. Leach, "Some experiments with a simple word recognition system," *IEEE Trans. Audio Electroacoust.*, vol. AU-16, pp. 256–261, June 1968.
[142] L. Shockey and R. Reddy, "Quantitative analysis of speech perception," in [41].
[143] J. Shoup, personal communication, 1975.
[144] H. F. Silverman and N. R. Dixon, "An objective parallel evaluator of segmentation/classification performance for multiple systems," *IEEE Trans. Acoust., Speech, Signal, Processing*, vol. ASSP-23, pp. 92–99, Feb. 1975.
[145] H. A. Simon and L. Siklossy, Eds., *Representation and Meaning*. Englewood Cliffs, NJ: Prentice-Hall, 1972.
[146] A. R. Smith, "A word hypothesizer for the Hearsay II speech system," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Philadelphia, PA, Apr. 1976.
[147] M. M. Sondhi, "New methods of pitch extraction," *IEEE Trans. Audio Electroacoust.*, vol. AU-16, pp. 262–266, 1968.
[148] C. C. Tappert, M. R. Dixon, A. S. Rabinowitz, and W. D. Chapman, "Automatic recognition of continuous speech utilizing dynamic segmentation, dual classification, sequential decoding and error recovery," Rome Air Development Center, Griffiss AFB, Rome, NY, Tech. Rep. TR-71-146, 1971.
[149] C. C. Tappert, N. R. Dixon, and A. S. Rabinowitz, "Application of sequential decoding for converting phonetic to graphic representation in automatic recognition of continuous speech (ARCS)," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 225–228, June 1973.
[150] C. C. Tappert, "Experiments with a tree-search method for converting noisy phonetic representation into standard orthography," *IEEE Trans. Acoust., Speech, Signal, Processing*, vol. ASSP-23, pp. 129–135, Feb. 1975.
[151] V. M. Velichiko and N. G. Zagoruiko, "Automatic recognition of 200 words," *Int. J. Man-Machine Studies*, vol. 2, pp. 223–234, 1970.
[152] P. Vicens, "Aspects of speech recognition by computer," Ph.D. dissertation, Comput. Sci. Dep., Stanford Univ., Stanford, CA, 1969.
[153] G. Ya Vysotsky, B. N. Rudnyy, V. N. Trunin-Donskoy, and G. I. Tsemel, "Experiment in voice control of computers," *Izv. Akad. Nauk SSSR, Tekn. Kibern.*, no. 2, pp. 134–143, 1970.
[154] T. G. Von Keller, "On-line recognition system for spoken digits," *J. Acoust. Soc. Amer.*, vol. 49, pp. 1288–1296, 1971.
[155] H. Wakita, "An approach to vowel normalization," Speech Communication Res. Labs., Santa Barbara, CA, Tech. Rep., 1975.
[156] D. E. Walker, W. H. Paxton, J. J. Robinson, G. G. Hendrix, B. G. Deutsch, and A. E. Robinson, "Speech understanding research annual report," Artificial Intelligence Center, Stanford Res. Inst., Menlo Park, CA, Project 3804, 1975.
[157] D. E. Walker, "The SRI speech understanding system," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 397–416, Oct. 1975.
[158] Walthen-Dunn, Ed., *Models for the Perception of Speech and Visual Form*. Cambridge, MA: M.I.T. Press, 1967.
[159] C. J. Weinstein, S. S. McCandless, L. F. Mondshein, and V. W. Zue, "A system for acoustic-phonetic analysis of continuous speech," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 54–67, Feb. 1975.
[160] C. Wherry, "VRAS: Voice recognition and synthesis," Naval Air Development Center, 1975.
[161] G. M. White and R. B. Neely, "Speech recognition experiments with linear prediction, bandpass filtering, and dynamic programming," in *Proc. 2nd USA–Japan Computer Conf.*, Tokyo Japan, Aug. 1975. Also to appear in *IEEE Trans. Acoust., Speech, Signal Processing*.
[162] R. A. Wiesen and J. W. Forgie, "An evaluation of the Lincoln Laboratory speech recognition system," *J. Acoust. Soc. Amer.*, vol. 56, S27 (A), 1974.
[163] J. J. Wolf, "Speech recognition and understanding," in *Pattern Recognition*, K. S. Fu, Ed. New York: Springer, 1975.
[164] W. A. Woods, "Transition network grammars for natural language analysis," *Commun. Ass. Comput. Mach.*, vol. 13, no. 10, pp. 591–602, 1970.
[165] ——, "Proposal for research on English language and speech understanding," Bolt Beranek and Newman, Inc., 1975.
[166] ——, "Motivation and overview of SPEECHLIS: An experimental prototype for speech understanding research," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 2–10, 1975.
[167] ——, "Syntax, semantics and speech," in [121, pp. 345–400].
[168] W. A. Woods and J. Makhoul, "Mechanical inference problems in continuous speech understanding," *Artificial Intelligence*, vol. 5, pp. 73–91, 1974.
[169] W. A. Woods, M. A. Bates, B. C. Bruce, J. J. Colarusso, C. C. Cook, L. Gould, J. I. Makhoul, B. L. Nash-Webber, R. M. Schwartz, and J. J. Wolf, "Speech understanding research at BBN, Final report on natural communication with computers," vol. I, Bolt Beranek and Newman, Inc., Rep. 2976, 1974.
[170] B. Yegnanarayana, "Effect of noise and distortion in speech on parametric extraction," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Philadelphia, PA, Apr. 1976.
[171] N. G. Zagoruiko, "Automatic recognition of speech," Quart. Progr. Status Rep., Speech Transmission Labs., KTH, Stockholm, Sweden, pp. 39–49, Jan–Mar. 1970.
[172] L. R. Bahl and F. Jelinek, "Decoding for channels with insertions, deletions, and substitutions with applications to speech recognition," *IEEE Trans. Inform. Theory*, vol. IT-21, pp. 404–411, 1975.
[173] L. R. Bahl, J. K. Baker, P. S. Cohen, N. R. Dixon, F. Jelinek, R. L. Mercer, and H. F. Silverman, "Experiments in continuous speech recognition," to appear in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Philadelphia, PA, Apr. 1976.
[174] N. R. Dixon and C. C. Tappert, "Toward objective phonetic transcription—An on-line interactive technique for machine-processed speech data," *IEEE Trans. Man-Mach. Syst.*, vol. MMS-11, pp. 202–210, 1970.
[175] H. F. Silverman and N. R. Dixon, "A parametrically-controlled spectral analysis system for speech," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-22, pp. 362–381, 1974.